Defense Threat Reduction Agency
8725 John J. Kingman Road, MS
6201 Fort Belvoir, VA 22060-6201

DTRA-TR-15-81

TECHNICAL REPORT

Robust Network Architecture Against Random Threats in WMD Environments: Theoretical Limits and Recovery Strategies

Distribution Statement A. Approved for public release, distribution is unlimited.

August 2017

HDTRA1-08-1-0024

Wenye Wang and
Hamid Krim

Prepared by:
North Carolina State University
Raleigh, NC 27695

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE *(DD-MM-YYYY)* 00-08-2017	**2. REPORT TYPE** Technical	**3. DATES COVERED** *(From - To)* April 15, 2008 - July 31, 2014

4. TITLE AND SUBTITLE

Robust Network Architecture Against Random Threats in WMD Environments:
Theoretical Limits and Recovery Strategies

5a. CONTRACT NUMBER
HDTRA1-08-1-0024

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Wenye Wang and Hamid Krim

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

North Carolina State University
Raleigh, NC 27695

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Threat Reduction Agency
8725 John J. Kingman Road, STOP 6201
Fort Belvoir, VA 22060-6201

10. SPONSOR/MONITOR'S ACRONYM(S)

DTRA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
DTRA-TR-15-81

12. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution Statement A. Approved for public release, distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This project presents a comprehensive summary of the problems under study, our approaches, technical con-tributions, and accomplishments that are supported throughout the entire project period, April 15, 2008 to July 31, 2014. There are two time periods for this project, one is the original award from September 1 2008 to August 31, 2011, and the other from September 1, 2011 to July 31, 2014 as the supplemental period. In the original project, we addressed two thrusts, namely network vulnerabilities and recovery strategies in the aftermath of WMD attacks. More specifically, this research targets a set of fundamental issues to understand network re-sponse following an attack by WMD/WME: how to model a network topology in the presence of attacks/failures from random threats; how to estimate or predict network survivability to sustain critical applications; how to de-sign/form network architecture to approach the theoretical limits of network robustness; and how to inter-operate with other available/limited sources for fast communication recovery.

15. SUBJECT TERMS

Network responses, physical networks, correlated failures, inter-operation, inter-dependent networks, topological analysis

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Paul Tandy
a. REPORT Unclassified	**b. ABSTRACT** Unclassified	**c. THIS PAGE** Unclassified	SAR	147	**19b. TELEPHONE NUMBER** *(include area code)* 703-767-4097

UNIT CONVERSION TABLE

U.S. customary units to and from international units of measurement[*]

U.S. Customary Units	Multiply by → ← Divide by[†]	International Units
Length/Area/Volume		
inch (in)	2.54×10^{-2}	meter (m)
foot (ft)	3.048×10^{-1}	meter (m)
yard (yd)	9.144×10^{-1}	meter (m)
mile (mi, international)	$1.609\,344 \times 10^{3}$	meter (m)
mile (nmi, nautical, U.S.)	1.852×10^{3}	meter (m)
barn (b)	1×10^{-28}	square meter (m^2)
gallon (gal, U.S. liquid)	$3.785\,412 \times 10^{-3}$	cubic meter (m^3)
cubic foot (ft^3)	$2.831\,685 \times 10^{-2}$	cubic meter (m^3)
Mass/Density		
pound (lb)	$4.535\,924 \times 10^{-1}$	kilogram (kg)
unified atomic mass unit (amu)	$1.660\,539 \times 10^{-27}$	kilogram (kg)
pound-mass per cubic foot (lb ft^{-3})	$1.601\,846 \times 10^{1}$	kilogram per cubic meter (kg m^{-3})
pound-force (lbf avoirdupois)	4.448 222	newton (N)
Energy/Work/Power		
electron volt (eV)	$1.602\,177 \times 10^{-19}$	joule (J)
erg	1×10^{-7}	joule (J)
kiloton (kt) (TNT equivalent)	4.184×10^{12}	joule (J)
British thermal unit (Btu) (thermochemical)	$1.054\,350 \times 10^{3}$	joule (J)
foot-pound-force (ft lbf)	1.355 818	joule (J)
calorie (cal) (thermochemical)	4.184	joule (J)
Pressure		
atmosphere (atm)	$1.013\,250 \times 10^{5}$	pascal (Pa)
pound force per square inch (psi)	$6.984\,757 \times 10^{3}$	pascal (Pa)
Temperature		
degree Fahrenheit (^{o}F)	$[T(^{o}F) - 32]/1.8$	degree Celsius (^{o}C)
degree Fahrenheit (^{o}F)	$[T(^{o}F) + 459.67]/1.8$	kelvin (K)
Radiation		
curie (Ci) [activity of radionuclides]	3.7×10^{10}	per second (s^{-1}) [becquerel (Bq)]
roentgen (R) [air exposure]	$2.579\,760 \times 10^{-4}$	coulomb per kilogram (C kg^{-1})
rad [absorbed dose]	1×10^{-2}	joule per kilogram (J kg^{-1}) [gray (Gy)]
rem [equivalent and effective dose]	1×10^{-2}	joule per kilogram (J kg^{-1}) [sievert (Sv)]

[*] Specific details regarding the implementation of SI units may be viewed at http://www.bipm.org/en/si/.

[†] Multiply the U.S. customary unit by the factor to get the international unit. Divide the international unit by the factor to get the U.S. customary unit.

2015-11-16

Robust Network Architecture Against Random Threats in WMD Environments: Theoretical Limits and Recovery Strategies

FINAL REPORT: Award No. HDTRA1-08-1-0024
(J-Understanding Target Network Response to WMD Attack)
Time Period: April 15, 2008 - July 31, 2014

Principal Investigator

Wenye Wang, Professor
Electrical and Computer Engineering Dept.
North Carolina State University
Raleigh, NC 27695
(919) 513 2549 (tel)
(919) 515 5523 (fax)
Email: wwang@ncsu.edu

Co-Principal Investigator

Hamid Krim, Professor
Electrical and Computer Engineering Dept.
North Carolina State University
Raleigh, NC 27695
(919) 513 2270 (tel)
(919) 515 2285 (fax)
Email: ahk@ncsu.edu

ABSTRACT

This project presents a comprehensive summary of the problems under study, our approaches, technical contributions, and accomplishments that are supported throughout the entire project period, April 15, 2008 to July 31, 2014. There are two time periods for this project, one is the original award from September 1 2008 to August 31, 2011, and the other from September 1, 2011 to July 31, 2014 as the supplemental period. In the original project, we addressed two thrusts, namely *network vulnerabilities* and *recovery strategies* in the aftermath of WMD attacks. More specifically, this research targets a set of fundamental issues to understand network response following an attack by WMD/WME: how to model a network topology in the presence of attacks/failures from random threats; how to estimate or predict network survivability to sustain critical applications; how to design/form network architecture to approach the theoretical limits of network robustness; and how to inter-operate with other available/limited sources for fast communication recovery.

With the increasing attention on the national infrastructure, such as civilian and military telecommunication networks, power grids, and transportation systems, these large-scale, inter-connected networks are vulnerable to WMD attacks. Under such attacks on communication media and facilities, the intrinsic nature of *networking* inevitably surrenders a given infrastructure to cascading or correlated failures in both temporal and spatial domains, which in turn, have a great and potentially devastating impact on network operation availability and performance. Therefore, we proposed to conduct a *two-year optional extension* during which we focused on network vulnerability to such failures, and on assessing network availability, subject to variations in traffic and user demands, as well as critical elements. In particular, we proposed to explore correlated failures, and their propagation with respect to temporal and spatial domain, in both *inhomogeneous networks* and *physical systems*.

To this end, we first studied the underlying network topology that is used to monitor, detect, and identify attacks, by analyzing the topological structure (i.e., operational state) of a network such that can i) very efficiently detect and localize failures; [1] ii) further address the so-called *dynamic failures*, and more specifically, correlated failures, to not only achieve *their early detection and classification* but to also *track them* and proceed to *their isolation* and all this, as proposed below, by a distributed computational strategy which hence easily scales with the network size. We then subsequently explored structurally how *cascading* and *correlated* failures are formed and will propagate in networks, and how such failures disrupt communications. We further moved on to the analysis and evaluation of *physical networks* by extending our models to smart grids, which are also known as the next generation power grid. In short, our proposed work included rapidly detecting, measuring, and tracking/predicting correlated failures in both time and spatial domains, as well as optimal design against such failures for inhomogeneous networks and their applications in physical networks.

As a result, our efforts advance the knowledge and fundamental understanding of the of WMD attacks, and the resulting failures in the infrastructure networks which form the backbone of civilian and military capabilities. Furthermore, we believe that our results will provide significant insights on the scope of damage due to cascading and correlated failures, including catastrophic loss of connectivity, unsuccessful missions, slow-down of the Internet, and damage to the economy and society at large. All together, the proposed research will leverage DoD capability in response to attacks from WMD/WME.

[1] This is more than an order of magnitude more efficient than any previous approach.

Contents

1 Objectives and Status of Efforts

Timely and accurate gathering and dissemination of information following an attack by Weapons of Mass Destruction (WMD) or Weapons of Mass Effect (WME) are crucial to preserving DoD military capabilities on the substratum of national infrastructure networks and military tactical networks. The destruction of infrastructure, combat and tactical networks, will be an immediate result from WMD stressors which can be biological, chemical, and nuclear weapons. For example, widely distributed failures of electronics from a nuclear electromagnetic pulses or a long-term denial of network elements or segments may be a consequence of WMD contamination. In addition to the impact of WMD stressors on the national network infrastructure, the impact of nuclear electromagnetic pulse on wireless networks may even be much more severe, as radio channels are open medium for wireless communications. This is in part due to wireless networks having become an indispensable element in civilian and military communications, and being almost exclusively the sole means of communication during a disaster relief and/or battlefield settings. For military applications, relying on centralized systems with base stations and on an established network is simply not even an option in light of typically hostile and dynamic (probably even unknown) environments. To overcome the limited radio transmission ranges (i.e., wireless devices can only communicate with others within their transmission range), nodes are equipped with an ability to forward information on behalf of others, i.e., multihop communications, which led to the research and development of *ad hoc networks* and have witnessed tremendous evolution in recent years. In addition, wireless ad hoc networks may take on a role of temporary replacement or supplement of the fixed infrastructure in reacting to failures and dynamic networking environments or applications.

Therefore, in the *original* project, we addressed two thrusts, namely *network vulnerabilities* and *recovery strategies* in the aftermath of WMD attacks. More specifically, this research targets a set of fundamental issues to understand network response following an attack by WMD/WME: how to model a network topology in the presence of attacks/failures from random threats; how to estimate or predict network survivability to sustain critical applications; how to design/form network architecture to approach the theoretical limits of network robustness; and how to inter-operate with other available/limited sources for fast communication recovery. That means, we have the following objectives:

- We aim to develop *new analytical models* in order to capture the impact of multiple failures due to random threats (aforementioned) on network dynamics in the face of WMD/WME disruption, such as connection (link) status, connectivity, and topology.
- We aim to design and analyze *new metrics* that it can characterize and estimate the impact of interdependence of a multitude of failures regarding network responses, based on either peacetime data or simulated threats.
- We aim to develop *new approaches* that re-form or self-heal a network architecture given complete or incomplete knowledge of communication environments and function of other nodes (Note that we use nodes for both people and wireless devices in this context), e.g., cooperative nodes or non-cooperative neighboring nodes whose cooperative functions may be disabled by energy depletion or WMD stressors.

- We aim to design *new distributed methods* that are distributed and with an ability to recover communications in heterogeneous networking environments by utilizing resources from other networks (sensor networks and back-haul networks) under the circumstances with limited connection or outage due to unknown causes. By *distributed*, we mean that the designed solutions should enable each node to work independently with others, without requiring a centralized controller.

The first two objectives are under the thrust of understanding *network response* to WMD stressors, with respect to the rules and parameters that govern network vulnerability and survivability. The last two objectives are under the thrust of *recovery strategies* to re-form a network and to inter-operate with other networks. In combination, we tackle the problem of identifying fundamental principles that govern network responses, and of facilitating robust, tactical communication networks under random threats.

With the increasing understanding of network responses to WMD failures and their impact, it is more and more clear that our national infrastructure, such as civilian and military telecommunication networks, power grids, and transportation systems are vulnerable to large-scale attacks in WMD environments. Under such attacks on communication media and facilities, the intrinsic nature of *networking* inevitably surrenders a given infrastructure to cascading or correlated failures in both temporal and spatial domains, which in turn, have a great and potentially devastating impact on network operation availability and performance. In the extension period of this project, we focus on network vulnerability to such failures, and on assessing network availability, subject to variations in traffic and user demands, as well as critical elements.

We note that there has been very limited work on the impact of tempo-spatially correlated and dynamic failures which are quite unique in networks subjected to WMD environments. It is in fact and in part, due to the lack of sound mathematical models of such failures, and of understanding these arising complex issues in networks which cannot easily be described in terms of graphs and methodologies for simple, homogeneous networks, as well as of predicting network inter-dependent responses. Consequently, there exists a critical gap in the knowledge required for research in mitigating network vulnerabilities and their potentially dire consequences. Towards reducing this gap, we propose to study the following issues in the expansion period of time:

(1) how to model and analyze correlated failures in temporal and spatial domains, with respect to their formation, propagation, and evolution capability?

(2) how to measure or predict network vulnerabilities in an inter-dependent networks? For instance, what is the temporal correlation between a cyber attack and a physical fault in power grid?

(3) how to design fast and distributed algorithms to ensure network security by detecting, localizing, and tracking failures? We focused our work on sensor networks, which have wide range of applications including surveillance and monitoring of potentially hazardous and inaccessible regions.

Status of Efforts: Through our productive research in the past five years, we have achieved significant research results toward the understanding and construction of robust network architecture to resist WMD/WME threats. As the first step of our research efforts, we have designed a novel node behavior model to accurately capture the behavior transitions of a node under WMD attacks, such as mobile, cooperative, faulty, destructive, and dead. The characterization of individual node behaviors lays the foundation for our advanced studies on the network structures, based on this we have further estimated the network survivability and communication feasibility in

WMD occurrences and designed a PROActive routing protocol to maintain network performance with tolerance of multiple node failures. Moreover, we have also thoroughly investigated the network counter-failure capability and the failure notification promptness by defining the novel concepts of network devolution process and information propagation speed. In addition, a new topological analysis approach has been used to detect failures in data space, which has the advantages of early detection and accurate localization of failures.

In the extended period of time, we also (i) designed of a low complexity distributed algorithm for detecting and tracking such failures. We assume that nodes inside the failure region are either destroyed or unable to communicate with any other node. The algorithm presented here does not assume any co-ordinate information for the nodes. We evaluate the algorithm using simulations; (ii) studied of correlated failures by considering constant and generic impact radius and their distribution, focusing on the persistent failures, which is one more step further than the one-time failure we studied earlier; (iii) investigated the impact of cyber-attack, focusing on the CSMA-CA based networks, which is one of the most widely used access technique in communication networks. In other words, the results of our study can be applied to a broad range of networks of the infrastructure networks. The key contribution of our work is that we *quantify* the impact of attackers' gain and thus find out what attacks would generate the most harmful disruption to the network, in contrast to many prior studies that are focused on the qualitative description and justification; (iv) studied the *fast tracking failures and identifying correlated failures* in power grid. As a result, we are able to track the failure without the need for any requirements on node density; and (v) developed a *Greenbench* that integrates power grid simulator PSCAD and networking simulator OMNeT++. In this simulator, we also designed three *data-centric* attacks and studied cascading failures in the power grids due to cyber attacks.

In this *final* report, we present the research issues under study, our approaches and technical contributions in the following *five* categories:

1. **Mobility modeling and analysis of coverage properties in mobile networks.**
2. **Network connectivity and vulnerability: modeling, analysis, and countermeasure.**
3. **Detection, localization, and tracking of systematic failures.**
4. **Correlated failures and their propagation in inhomogeneous networks.**
5. **Cascading Failures in Power Grids due to Communication and Cyber Attacks.**

Specifically, we plan to focus on: (i) The design of a PROActive routing protocol which is able to avoid delivering data to destructive or faulty nodes so as to maintain network operation and performance in the presence of multiple failures; (ii) Under random failures, the fundamental understanding of network topology devolution and transitions; (iii) Given the occurrence of a failure, the speed of information propagation in large-scale networks; and (iv) a new approach to characterize network data through topological analysis. Among four topics under study, the first one is in the thrust of design robust network architecture against failures and attacks, while the last three topics are in the thrust of under- standing of network responses to failures with respect to fundamental limitations.

2 Mobility Modeling and Analysis of Coverage Properties

2.1 Semi-Markov Smooth (SMS) Mobility Model

Node mobility, by and large, describes the presence of a node in a network, is a key factor that defines network architecture, especially in multi-hop wireless networks. By mobility, a node can move around and thus yielding node to node radio links for communications. During the course of studying mobility-induced failures, we have found that existing mobility models do not have the desired properties for our analysis. We are motivated to design a new mobility model that can abide by the physical law of moving objects to avoid abrupt moving behaviors, and can provide a *microscopic view* of mobility such that node mobility is controllable and adaptive to different network environments. In summary, this model is expected to *unify* the desired features as follows:

1. Smooth and sound movements: A mobility model should have temporal features, i.e., a mobile node's current velocity is dependent on its moving history so that smooth movements can be provided and mobile nodes should move at stable speed without the average speed decay problem [1].

2. Consistency with the physical law of a smooth motion: In order to mimic the kinetic correlation between consecutive velocities in a microscopic level of a node, a mobility model should be consistent with the physical law of a smooth motion in which there exists acceleration to start, stable motion and deceleration to stop for controllable mobility [2, 3].

3. Uniform nodal distribution: As most of analytical studies of MANETs are based on the assumption of uniform nodal distribution, such as network capacity and delay [4], network connectivity, topology control [5] and link change rate [6], a mobility model should generate uniform spatial node distribution. Otherwise, the non-uniform node distribution caused by a mobility model may invoke misleading information and results [7].

4. Adaptation to diverse network application scenarios: In order to properly support rich MANET applications having complex node mobility and network environments, such as group mobility and geographic restriction, a generic mobility model which is adaptive to different mobility patterns is highly desirable. TABLE 1 illustrates a detail comparison based on properties of current typical mobility models and those of our proposed SMS model, where independent mobility parameters: *speed* (V), *movement duration* (T), *destination* (D) and *direction* (θ) with respect to different mobility patterns are also included.

2.1.1 Model Description

Based on the physical law of a smooth motion, a movement in the SMS model contains three consecutive moving phase: *Speed Up* phase, *Middle Smooth* phase, and *Slow Down* phase, respectively. After each movement, a mobile node may stay for a random pause time.

- **Speed Up Phase (α–Phase)**

Table 1: Properties of Different Mobility Models.

Attributes	RW [8]	RWP [9]	RD [7]	SR [10]	GM [11]	SMS
Parameters	V, θ	V, D	V, θ, T	V, θ	V, θ	V, θ, T
Movement Phases	{moving, pause}	{moving, pause}	{moving, pause}	{moving, pause}	{moving}	{speed-up, middle smooth, slow-down, pause}
Smoothness	No	No	No	Yes	Yes	Yes
Speed Decay	May	Yes	No	No	No	No
Uniform Node Distribution	close	no	yes	close	yes	yes
Mobility Scale	macroscopic	macroscopic	macroscopic	microscopic	microscopic	microscopic
Unified Model	No	No	No	No	No	Yes
Controllability	low	Low	Low	Medium	Medium	High

For every movement, an object needs to accelerate its speed before reaching a stable speed. During time interval $[t_0, t_\alpha] = [t_0, t_0 + \alpha\Delta t]$, an SMS node travels with α time steps. At initial time t_0, the node randomly selects a *target speed* $v_\alpha \in [v_{\min}, v_{\max}]$, a *target direction,* $\phi_\alpha \in [0, 2\pi]$, and the total number of *time steps* $\alpha \in [\alpha_{min}, \alpha_{\max}]$. These three random variables are independently uniformly distributed.

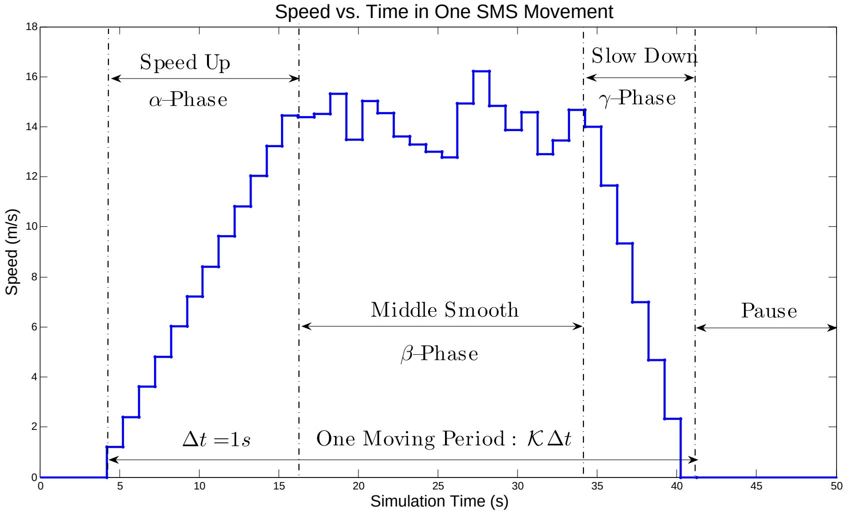


Figure 1: An example of speed vs. time in one SMS movement.

- **Middle Smooth Phase (β–Phase)**

In reality, after the speed acceleration, a moving object should have a smooth motion according to its stable velocity. Correspondingly, once the node transits into β–phase at time t_α, it randomly selects β time steps to determine the middle smooth (β–phase) duration interval: $(t_\alpha, t_\beta] = (t_\alpha, t_\alpha + \beta\Delta t]$. Where β is uniformly distributed over $[\beta_{min}, \beta_{\max}]$. Within β–phase, the mobility pattern at each time step is similar to what is defined in Gauss Markov (GM) model [11].

- **Slow Down Phase (γ–Phase)**

  In real-life, every moving object needs to reduce its speed to zero before a full stop. In order to avoid the sudden stop event happening in the SMS model, we consider that the SMS node experiences a slow down phase to end one movement. In detail, once the node transits into slow down (γ–phase), at time t_β, it randomly selects γ time steps and a direction $\phi_\gamma \in [0, 2\pi]$. Where γ is uniformly distributed over $[\gamma_{min}, \gamma_{\max}]$. In γ–phase, the node evenly decelerates its speed from v_β, the ending speed of β–phase, to $v_\gamma = 0$ during γ time steps.

2.1.2 Semi-Markov Process of SMS Model

We consider *pause* as another phase, then the stochastic process of SMS model is described as an iterative four-state transition process. Let I denote the set of phases in an SMS movement, then $I(t)$ denotes the phase of SMS process at time t, where $I = \{I_\alpha, I_\beta, I_\gamma, I_p\}$. Accordingly, $\{\mathcal{Z}(t); t \geq 0\}$ denotes the process which makes transitions among phases in the stochastic modeling of SMS movements. Since the transition time between consecutive moving phases (states), i.e., phase duration time, has discrete uniform distribution, instead of an exponential distribution, $\{\mathcal{Z}(t)\}$ is a *semi-Markov process* [12]. *This is the very reason that our mobility model is called Semi-Markov Smooth model* because it has an Semi-Markov process and it complies with the physical law with smooth movement. Let $\pi = (\pi_\alpha, \pi_\beta, \pi_\gamma, \pi_p)$ denote the time stationary distribution of SMS process. Then, the time stationary distribution for each phase of SMS model is:

$$\pi_m = \lim_{t \to \infty} Prob\{I(t) = I_m \in I\} = \frac{E\{T_m\}}{E\{T\} + E\{T_p\}}, \tag{1}$$

where $E\{T_m\}$ is the expected duration time of m–phase in an SMS movement. $E\{T\}$ and $E\{T_p\}$ are the expected SMS movement period and pause period, respectively. Specifically, $E\{T\} = E\{\alpha\Delta t\} + E\{\beta\Delta t\} + E\{\gamma\Delta t\}$. Since Δt is a constant unit time, for the sake of simplicity, Δt is normalized to 1 second in the rest of the paper.

2.1.3 Analysis of Steady-State Speed and Node Distribution

To generate stable nodal movements, a *sound* mobility model should select the speed independently from travel times [1], which is exactly what occurs in SMS model. Here, we evaluate the stochastic property of steady-state speed in SMS model and verify that SMS model *can* eliminate speed decay problem and achieve stable nodal movements. In order to find out whether there exists the speed decay phenomenon in SMS model, it is necessary to obtain both initial average speed $E\{v_{ini}\}$ and average steady-state speed $E\{v_{ss}\}$.

According to the initial stage, each node starts from an SMS phase with a certain state probability based on the time stationary distribution of the SMS process. The average speed in each moving phase of an SMS movement is obtained as: $E_{I_\alpha}\{v\} = E_{I_\gamma}\{v\} = \frac{1}{2}E_{I_\beta}\{v\} = \frac{1}{2}E\{v_\alpha\}$. Then we derive the CDF of steady-state speed $Pr\{V_{ss} \leq v\}$ can be derived from the limiting fraction of time when step speeds of a node are less than v, as the simulation time t approaches to infinity. Let $\mathcal{M}(t)$ and $\mathcal{M}_p(t)$ denote the total number of time steps that a node travels and pauses during $[0, t]$, respectively. Thus, $Pr\{V_{ss} \leq v\}$ is derived as:

$$Pr\{V_{ss} \leq v\} = \lim_{t\to\infty} \frac{\sum_{n=1}^{\mathcal{M}(t)} \mathbf{1}_{\{v_n \leq v\}} + \sum_{n=1}^{\mathcal{M}_p(t)} \mathbf{1}_{\{v_n \leq v\}}}{\mathcal{M}(t) + \mathcal{M}_p(t)}, \tag{2}$$

where $\mathbf{1}_{\{.\}}$ is the indicator function. Thus, if the event that $\{v_n \leq v\}$ is true, then $\mathbf{1}_{\{v_n \leq v\}} = 1$, otherwise $\mathbf{1}_{\{v_n \leq v\}} = 0$. Finally, we can obtain:

$$\begin{aligned} E\{v_{ss}\} &= E_{I_\alpha}\{v_{ss}\} + E_{I_\beta}\{v_{ss}\} + E_{I_\gamma}\{v_{ss}\} + E_{I_p}\{v_{ss}\} \\ &= \frac{\frac{1}{2}E\{v_\alpha\}(E\{\alpha\} + 2E\{\beta\} + E\{\gamma\})}{E\{T\} + E\{T_p\}}. \end{aligned} \tag{3}$$

We observed that the average initial speed is exactly same as average steady-state speed in SMS model, i.e., $E\{v_{ini}\} = E\{v_{ss}\}$. Therefore, we proved that SMS model does not have speed decay problem.

Since an SMS node selects direction, speed and phase time independently, SMS model can be considered as an enhanced random direction (RD) model with memorial and microscopic property on step speed and direction. RD model was proved to maintain uniform node distribution in [7]. Here, we want to prove that SMS model also yields uniform node distribution. We evenly distribute all mobile nodes in the simulation region at the initial time. For a simple representation, we normalize the size of the simulation region to $[0, 1)^2$. (X_j, Y_j), v_j, and ϕ_j denote the ending position, speed and direction in a node's j^{th} step of its first movement, respectively. When an SMS node reaches a boundary of the simulation region, it wraps around and reappears instantaneously at the opposite boundary in the same direction to avoid biased simulation results. Under this condition of border wrap, we have the following Lemma:

***Lemma* 1.** *In SMS model, if the initial position $\boldsymbol{P}(0)$ and the first target direction ϕ_α of a mobile node are chosen independently and uniformly distributed on $[0, 1)^2 \times [0, 2\pi)$ at time $t = 0$, then the location and direction of the node remain uniformly distributed all the time.*

Given that the initial position (X_0, Y_0) and ϕ_α of a node have independently uniform distribution, the joint probability of ending position and direction of the node's first step movement is:

$$\begin{aligned} & Pr(X_1 < x_1, Y_1 < y_1, \phi_1 < \theta) \\ &= Pr(X_1 < x_1 | \phi_1 < \theta) \cdot Pr(Y_1 < y_1 | \phi_1 < \theta) \cdot Pr(\phi_1 < \theta) \\ &= \frac{1}{2\pi} \int_{\phi_1=0}^{\theta} \Big(\int_{x_0=0}^{1} \mathbf{1}_{\{x_0 + v_1 \cos(\phi_1) - \lfloor x_0 + v_1 \cos(\phi_1) \rfloor < x_1\}} dx_0 \cdot \\ & \qquad \int_{y_0=0}^{1} \mathbf{1}_{\{y_0 + v_1 \sin(\phi_1) - \lfloor y_0 + v_1 \sin(\phi_1) \rfloor < y_1\}} dy_0 \Big) d\phi_1 \\ &= \frac{x_1 y_1 \theta}{2\pi}. \end{aligned} \tag{4}$$

The result in (4) shows that (X_1, Y_1) and ϕ_1 are uniformly distributed on $[0,1)^2 \times [0, 2\pi)$. Following the same methodology, by induction on each following step, Lemma 1 is proved. The detailed proof is described in [13].

2.1.4 Simulation Results and Model Comparisons

Here, we verify the above theoretical analysis of SMS model by simulations and compare the results with RWP and GM models. We integrate our SMS model into the *setdest* of ns-2 simulator, which currently provides both an original and a modified version of RWP model. In order to compare simulation results between RWP and SMS model, 1000 mobile nodes move in an area of $1401m \times 1401m$ during a time period of 1500 seconds. For a better demonstration, we simulated both the SMS model and the original RWP model with zero pause time. Both GM and SMS model set the time slot Δt as 1 second and the memory parameter ζ as 0.5, respectively. In SMS model, we consider the range of each moving phase duration time as $[6, 30]$ seconds.

- **Average Speed**

  Here, we are interested in comparing the average speed between SMS model and RWP model and validate our analytical proof shown in Section 2.1.3. To obtain the average node speed, we first calculate the average speed of each node within every 10 seconds, and then calculate the average speed among all the nodes. The corresponding numerical results of average speed vs. a time period of 1500 seconds are shown in Figure 2. Given the simulation condition of zero pause time and $E\{\alpha\} = E\{\beta\} = E\{\gamma\}$, from (3), the theoretical result of $E\{v_{ss}\}$ of SMS model is obtained as: $E\{v_{ss}\} = \frac{2}{3}E\{v_\alpha\} = 6.7$ m/sec. From Figure 2, we observe that the average speed of the SMS model is stable from the beginning of simulation at the value around 6.7 m/sec, which perfectly matches the theoretical result. Therefore, the simulation results validate our analytical conclusion that the average speed of SMS model does not decay over time. Whereas, the average speed of RWP model keeps on decreasing as the simulation time progresses, which is its well-known average speed decay problem [14].

- **Spatial Node Distribution**

  To verify our derivation of node distribution, we distribute nodes uniformly in the simulation region at the initial time. Then, we sample the node position at the 500^{th} second for SMS model, and the 1000^{th} second for both RWP and SMS models. A top view of two-dimensional spatial node position of RWP and SMS models are shown in Figure 3. The results of RWP model in Figure 3(a) show that the node density is the maximum at the center of the region, while it is almost zero near the network boundary, which agrees with the previous study [15]. In contrast, in Figures 3(b) and 3(c), the two node density samples of the SMS model at different time instants are similar and mobile nodes are evenly distributed in the simulation region. Since these two time instants are arbitrarily selected, we verified our proof that the SMS model with border wrap maintains uniform spatial node distribution over time.

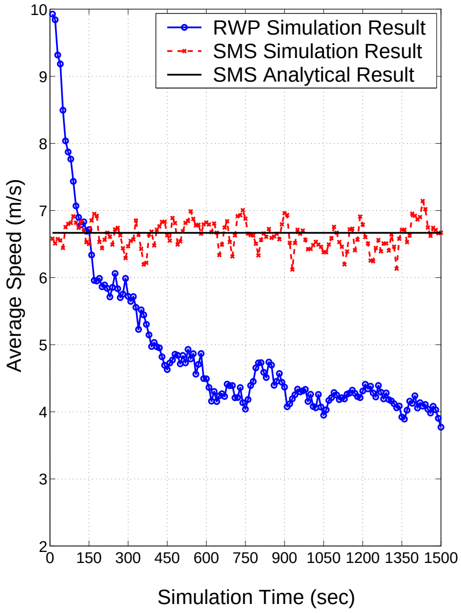
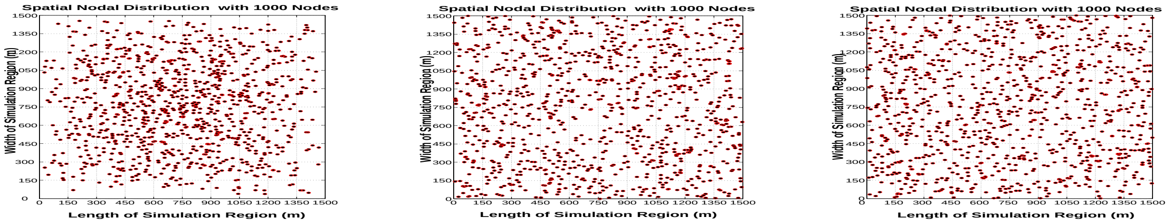


Figure 2: Average speed vs. simulation time.



(a) RWP 2-dimensional at the 1000^{th} sec. (b) SMS 2-dimensional at the 500^{th} sec. (c) SMS 2-dimensional at the 1000^{th} sec.

Figure 3: Top-View of node distribution of the RWP model at the 1000^{th} sec and the SMS model at the 500^{th} and the 1000^{th} sec, respectively.

2.1.5 Comparison

We also compare the simulation results of link lifetime distribution and average node degree among the RWP, GM and SMS models. We find that the probability mass function (PMF) of link lifetime of both SMS model and GM model decreases exponentially with time. In contrast, there is a peak at the 25^{th} second of link lifetime distribution in RWP model. Hence, it turns out that mobility models with macroscopic mobility pattern would have different link and path properties from those of mobility models with microscopic pattern, such as the GM and SMS models. Therefore, SMS model is more accurate for the simulation on link lifetime in MANET than other models. We find that the result obtained in RWP model is apparently larger than that in GM and SMS model. This is because the majority of nodes move into the center region in RWP model as the simulation time proceeds. That means the network connectivity evaluation based on RWP model could be over optimistic.

Therefore, SMS model with uniform node distribution is preferable for network connectivity study in MANET.

2.2 Coverage Analysis of Mobile Networks

In this part of the project, we investigated coverage properties of mobile networks. Consider the scenario where a set of mobiles nodes are following a certain mobility pattern in a region of interest. A natural problem which arises in these cases is the measurement of coverage properties over time of the network. Previous work in this area was concerned with "snap-shot" statistics such as average area covered at a given time or the time taken until every point in the region is covered at least once. But, statistics such as average time for which a coverage hole remain in the network until it disappears or merges with some other hole, have not been addressed in the literature. The later statistics were the target of this research, and we outline our results below.

2.2.1 Problem formulation

The coverage information of the mobile is summarized in a 'barcode' describing the birth and death times of homological features in the network over time, and we describe the relationship between these features and the coverage properties. The barcode is obtained by employing a method from the mathematical field of computational topology, called zigzag persistent homology [16]. Specifically, the network at each time t_i is modeled as a simplicial complex K_{t_i}, and the relation between the network at two consecutive times t_i and t_{i+1} is inferred through their inclusions into the union. This results in a "zig-zag" diagram as shown below:

$$\begin{tikzcd}
 & (K_{t_1} \cup K_{t_2}) & & (K_{t_2} \cup K_{t_3}) & & (K_{t_{T-1}} \cup K_{t_T}) & \\
K_{t_1} \arrow[ur] & & K_{t_2} \arrow[ul] \arrow[ur] & & \cdots \arrow[ul] \arrow[ur] & & K_{t_T} \arrow[ul]
\end{tikzcd} \tag{5}$$

This sequences of spaces and the inclusions maps in turn produce a sequence of homology spaces and linear maps as follows,

$$\begin{tikzcd}
 & H_1(K_{t_1} \cup K_{t_2}) & & H_1(K_{t_2} \cup K_{t_3}) & & H_1(K_{t_{T-1}} \cup K_{t_T}) & \\
H_1(K_{t_1}) \arrow[ur] & & H_1(K_{t_2}) \arrow[ul] \arrow[ur] & & \cdots \arrow[ul] \arrow[ur] & & H_1(K_{t_T}) \arrow[ul]
\end{tikzcd}$$

Briefly, topological features such as coverage holes at a given time are reflected in the dimension of the respective homology space, and their persistence over time may be observed through the analysis of the induced linear maps. The result of this analysis is summarized in a barcode, as shown for example in Figure 6, from which the statistical properties may be determined.

2.2.2 Results

We showed that zig-zag persistence may be used to perform the following tasks in a coordinate free setting:

1. Comparison of mobility patterns.

2. Coordinate free estimation of hole size.

3. Tracking individual coverage holes.

Comparison of mobility patterns The length of the barcode signifies the time for which a coverage hole in the network persists before being covered or merging with some other hole. Therefore, we would expect that the distribution of times of which these holes persist would vary with the mobility pattern. This was indeed the case, and as shown in Figure 4, we were able to distinguish very clearly between Discrete Brownian motion [17] and Straight Line motion [17, 18].

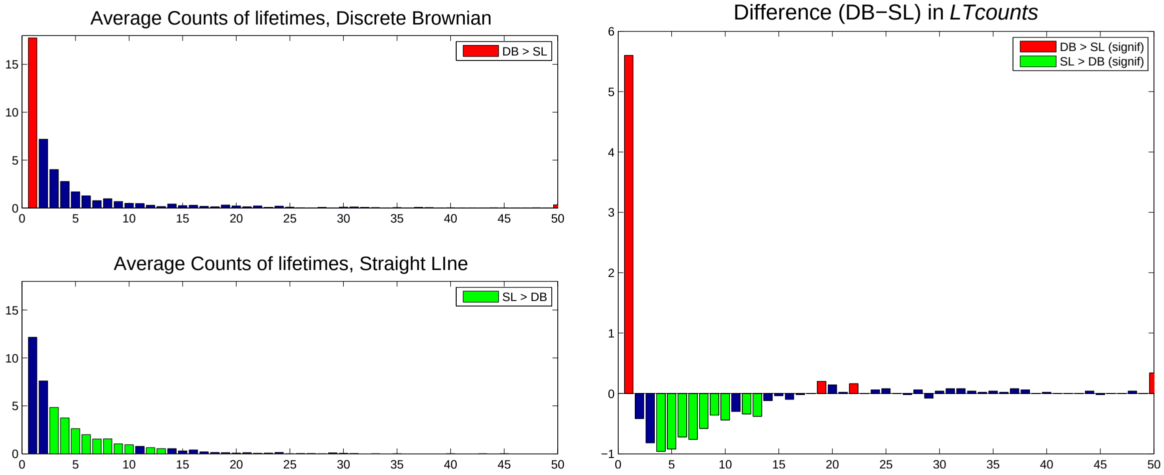


Figure 4: *LTcounts* for the Discrete Brownian (top left) and Straight Line (bottom left) mobility patterns, as well as the paired difference in *LTcounts* (right), with the lifetimes whose frequency has a statistically significant difference between the groups highlighted. The lifetimes that occur more frequently in the Discrete Brownian pattern ($t = 1, 19, 22$ and ,50) are highlighted in red in the top plot, and those that occur more frequently in the Straight Line pattern ($t = 4, \ldots, 13$) are highlighted in green in the bottom plot.

Coordinate free estimation of hole size The barcode obtained from zigzag persistence gives us a quantitative descriptor for the time-varying coverage of a network. However, the presence of a long bar in the barcode may or may not geometrically correspond to a large hole . Given that our network is described as a sequence of adjacency matrices (describing the simplicial complex at each snapshot, but without coordinate information), the best estimate available is the hop-length of the shortest cycle surrounding a hole. This can be obtained without having to compute the shortest cycle explicitly, by performing a hop-distance filtration on the simplicial complex (at each time point). In other words, given a cycle in the graph which surrounds a coverage hole, by successively adding edges between nodes which are multiple hops away, and checking whether the cycle is "filled in", we may obtain an estimate of the size of coverage hole by this approach. In practice, we observe a strong correlation between a hole size estimated this way with the actual geometric area as shown in Figure 5.

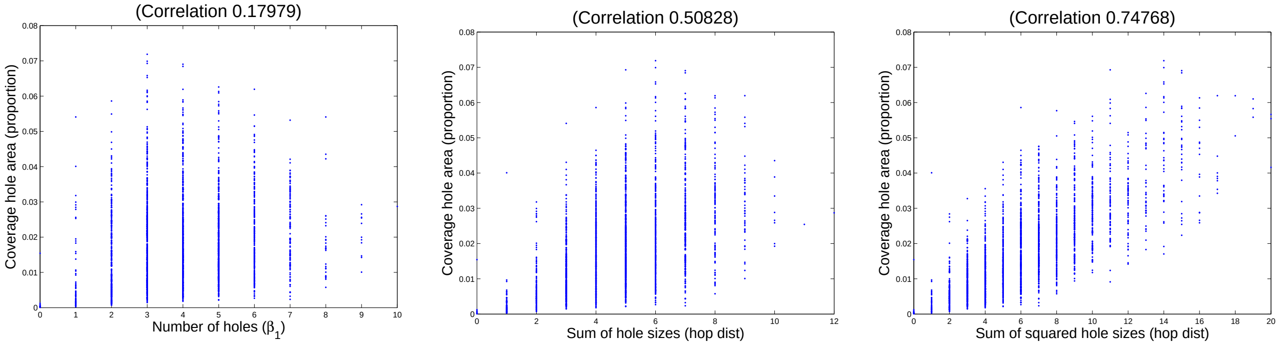


Figure 5: Relationship between *coverage hole area* (measured as proportion of total area) and various homological features. Left - first betti number $r = 0.176$), middle - sum of hole sizes (measured using depth in hop distance filtration, $r = 0.505$), right - sum of squared hole sizes (measured using depth in hop distance filtration, $r = 0.747$).

Tracking individual coverage holes Intuitively, our method aims to compute a 'canonical basis', i.e., a set of cycles, where there is one representative cycle surrounding each hole. Given the Rips complex for a static sensor network, without an embedding or geometric information, such a canonical basis is impossible to obtain. In the time-varying setting however, a small amount of 'canonical' information is available: when a coverage hole is first formed by the removal of a 2-simplex (triangle), the boundary of that triangle is known to surround exactly the hole of interest. The idea behind our method is then to use that boundary as the representative cycle for the homology class at its birth time, and propagate that information forward through the sequence of complexes as best as possible.

Figure 6 illustrates a network which is initially fully covered, and has a number of small coverage holes appearing over time, one of which is persistent. The barcode displaying lifetimes of homological features can be seen in the top left, with the bars color-coded to correspond to their associated representative cycles in the other figures. It can be seen that each representative cycle remains relatively tight around one coverage hole, and the set of cycles does correspond to a canonical basis at each time point. Overall, when a network is dense enough that its coverage holes appear and disappear in an isolated fashion (as opposed to splitting and merging with other holes), this method performs very well. Preliminary results have appeared in [19].

2.2.3 Summary

We have worked on developing an analysis of various aspects of networks under the effect of catastrophic failures such as those caused by WMDs. Starting with networks with static nodes and static topology, we have developed distributed and coordinate free algorithms to detect and localize failures. For networks with time-varying topology, we have shown that the boundary update, and the subsequent tracking of a spreading failure, may be

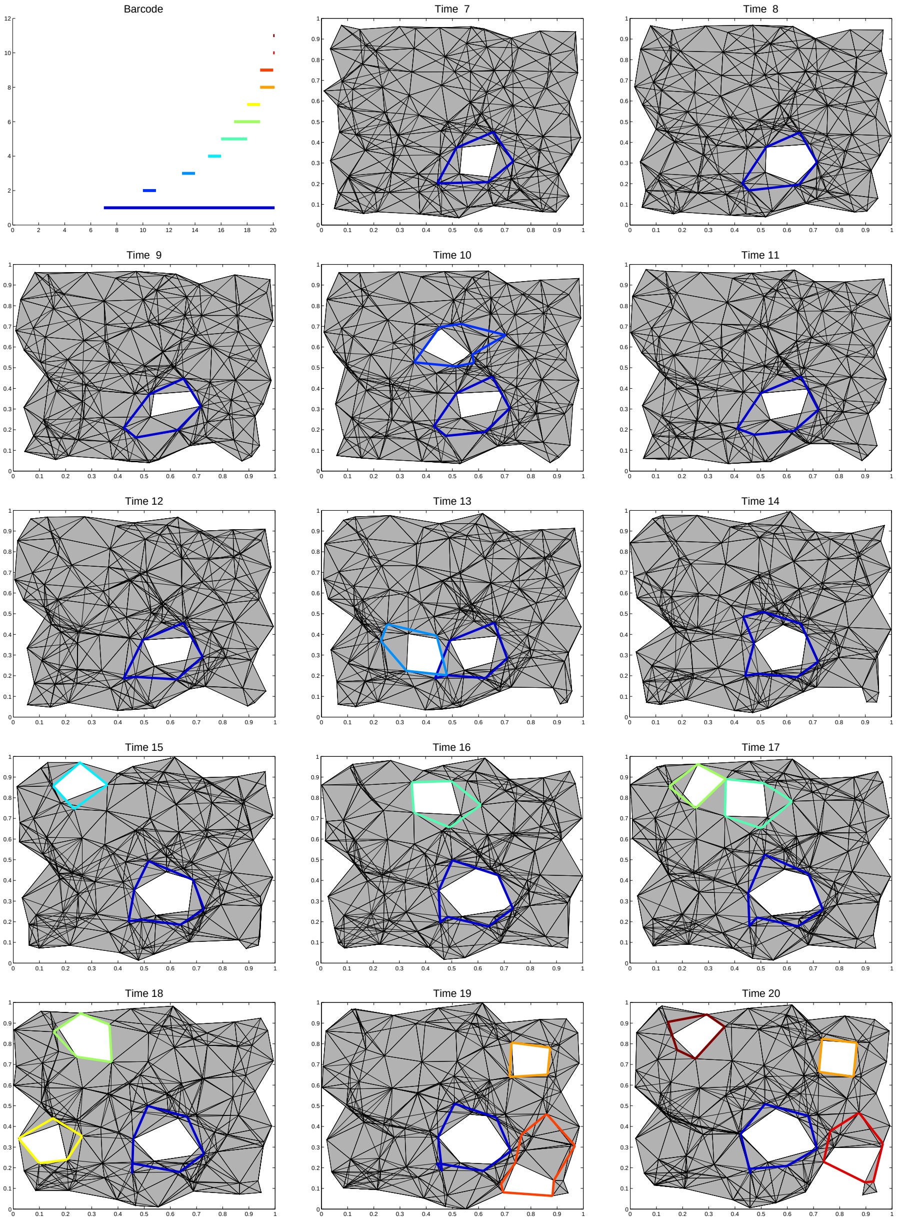


Figure 6: Representative cycles for the intervals obtained using zigzag persistence in a dense network, with color-coding between the representative cycle at each time point and its corresponding interval (barcode - top left).

performed using only the edge length information in real time. Building on our success in these scenarios, we further developed methodologies for quantifying and tracking the coverage in mobile networks (where the nodes themselves are also in motion).

This result demonstrated that the integration of traditional graph theory, distributed algorithms and the upcoming field of algebraic topology serves as an effective set of tools for robust analysis of networks. The focus of the funded effort herein, has focused on sensor networks, the nature of which endows certain geometric constraints to the construction of the network. On the other hand, there are many important networks in use which do not necessarily follow these geometric constraints and these networks are often interconnected and interdependent. In the future, we plan on building on the accomplished work and on what we have learnt from it, to develop algorithms and techniques which may address further issues in cyber-physical networks, where control issues have to also be carefully addressed.

3 Network Connectivity and Vulnerability: Modeling, Analysis, and Countermeasure

3.1 Understanding the Impact of Multi-Failure on Network Architecture

The impact of mobile node behaviors in ad hoc networks has been discussed in recent studies. For example, in [20] a distributed and scalable acceptance algorithm called *GTFT* was proposed to enable nodes to decide whether accept or reject relay requests in terms of cyber-attacks. In [21], DoS attacks launched by malicious nodes, *Jellyfish* and *Blackhole*, were shown to have a network partitioning effect that degrades the network performance. Recall that routing is the basic function in a wireless ad hoc network, every node action such as node movements, misbehavior, and attacks will ultimately affect routing procedures. For example, node mobility, which is the intrinsic feature of mobile ad hoc networks, may incur the routing table update and path re-selection. On the other hand, routing operations may be interrupted or distorted via node misbehavior: a selfish node may not forward packets for other nodes; a DoS attacking node may reorder or drop packets; and a failed node may not respond to route discovery messages. Thus, routing protocols cannot be performed correctly and the network is under the operation of *communication malfunction*. Therefore, we focus our efforts on the study of network survivability in the presence of multiple failures that due to node mobility, faulty communication, and dead nodes in the aftermath of attacks.

3.1.1 Characterization of Malfunction of Nodes and Operation

Basically, current modeling approaches and models have two major limitations; that is, memoryless property of each node's status, and independence of node failures under threats. First, most of these models are based on the assumption of exponential distribution of individual events, e.g., the transition time between two node failures follows an exponential distribution. Unfortunately, failure events, often times, do not follow exponential distributions in real-time distributed systems [22, 23, 24]. It is hence unable to reflect the correlation of multiple failures and their dependencies since a plain exponential distribution is memoryless. Second, most of these models can describe *single, independent failures* only, e.g., an energy model is used to estimate probability of power outage; mobility models are used to estimate link lifetime; traffic models are used to analyze throughput, and so on. In summary, existing modeling approaches fail to describe diverse, correlated, random threats in the face of WMD attacks, though they are appropriate to understand basic performance issues of wireless networks. A large number of potential failures are left out in these models, including connection breakage due to the combined effect of mobility, power outage, channel variability, incomplete knowledge of the operation capabilities of other nodes, and faulty network protocol behaviors.

3.1.2 Definition of Network Survivability

The survivability of a network refers to the ability that a network resists failures not only due to physical damages, operational errors or misconfiguration, but also due to adversaries. In order to enable a wireless ad hoc network in operation, it is necessary to keep the network connected, whenever practical. Hence, we employ *network connectivity* as the measure of the survivability for wireless ad hoc networks. In this work, we denote a wireless ad hoc network by $\mathcal{M}$ and its node set by $\mathcal{N}$. For a k-connected network $\mathcal{M}$, the maximum value of k is defined as the *connectivity* of $\mathcal{M}$, denoted by $\kappa(\mathcal{M})$ (see graph connectivity in [25]). Then we define the network survivability as follows:

Definition 1. *The* network survivability *of $\mathcal{M}$, denoted by $NS_{\mathcal{M}}(k, N)$, is defined as the conditional probability that $\mathcal{M}$ is k-connected conditional on the system size $|\mathcal{N}|$, that is*

$$NS_{\mathcal{M}}(k, N) = Pr(\kappa(\mathcal{M}) = k \mid |\mathcal{N}| = N), \tag{6}$$

where $|\mathcal{N}|$, the cardinality of $\mathcal{N}$, is a random variable and N is a value of the system size.

In this definition, network survivability is a probabilistic measure of network connectivity, and affected by multiple failures due to various causes in wireless ad hoc networks. The challenge in solving this problem is to take the communication malfunction into account instead of deriving the probability of node isolation as a resort to analyze network connectivity by some previous work [26, 27], where a node is isolated only because of no active neighbors. Our approach toward this problem is composed of three steps: *(i)* we study the evolution of node behaviors with potential routing malfunction, especially, we aim to find the stochastic properties of node behaviors, *(ii)* we investigate the node isolation problem due to the effect of abnormal routing operations, and (iii) based on the understanding of node isolation, we derive the network survivability.

3.1.3 Node Behavior Modeling

Our modeling of multiple failures is based on a rigorous classification of malfunction of nodes and network protocols as a result of random threats. We found that node behavior can be modeled by a *Semi-Markov Process* (SMP) to characterize transient and steady failures over time [28, 29]. We first associate nodes to one of four classes depending on their status,

- *Cooperative state (C)*: a node is said to be cooperative when it can correctly perform all designed communication/routing functions. This node status is *expected* during peacetime without WMD attacks or stressors.
- *Faulty state (F)*: a node is said to be faulty if this node cannot correctly perform all functions, which can be a result of power depletion and/or electronic troubles. However, these nodes do not initiate or trigger attacks to others, even though they may cause cascading failures, e.g., one node cannot forward information

on time, making its neighbors unable to correctly receive data and may further make excessive requests so as to jam radio links.

- *Destructive state (D)*: a node is said to be destructive if it appears to be cooperative (e.g., implement control messages correctly), but in fact, it interrupts normal communication functions, by delaying, reordering or dropping packets, or even sending fake routing messages and other faulty network protocol behaviors , such as launching denial-of-service (DoS) attacks [30].
- *Inactive state (I)*: a node is said to become inactive or failed if it cannot participate in any communication with others. This may be a result of a dead device, or being isolated from others due to a node moving out the transmission range of others, i.e., the effect of node mobility. This type of nodes, however, would not launch any attacks.

The nodes' status are thus described according to their functions subject to many random threats. Let $\mathcal{S} = \{C, F, D, I\}$ be the state space with the following properties: (1) A node in state i will enter state j with probability p_{ij}; and (2) given that the next state to be entered is state j, the transition time from states i to j ($i, j \in \mathcal{S}$) follows a general distribution $F_{ij}(t)$. Therefore, the node behavior model can be better described as a *Semi-Markov Process* (SMP), denoted by $Z(t), t \geq 0$, with a matrix of transition functions $Q_{ij}(t) = p_{ij} \cdot F_{ij}(t) \;\; i, j \in S$ [28]. With state space $\mathcal{S}$, a discrete time Markov chain (DTMC), denoted by $X_n, n \geq 0$ can be constructed with transition probability matrix $\mathbb{P} = (p_{ij})$, which is the *embedded* Markov chain of the Semi-Markov process $Z(t)$.

Figure 7 depicts the node behavior model defined above.

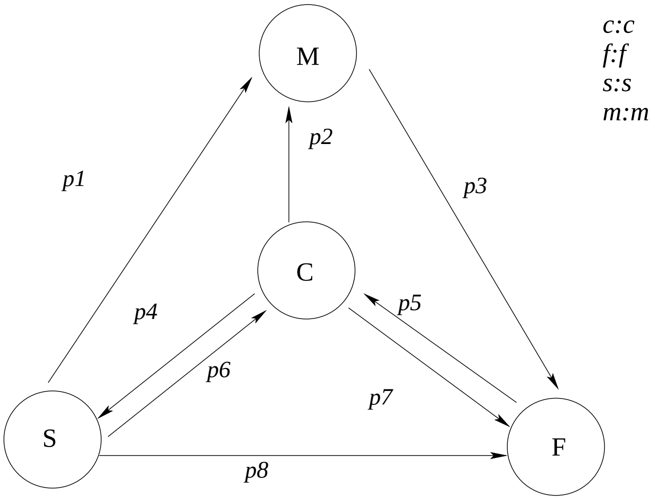


Figure 7: The semi-Markov process for node behavior evolution.

- **Stochastic Properties of Node Behavior Model.**

  In particular, we are interested in the probability that $Z(t)$ is in a certain state i, i.e., $P_i \triangleq \lim_{t\to\infty} P(Z(t) = i|Z(0) = j)$. Nevertheless, the existence of the limiting distribution needs to be verified.

  From Figure 7, we can see that the *embedded* Markov chain of $Z(t)$, denoted by X_n, has a finite state space $\mathcal{S}$, and in X_n each state can reach other states within finite steps and itself within one step. Thus, X_n is *irreducible* and *ergodic*. By *Corollary 9-1 (pp. 325)* in [12], we know that $Z(t)$ is *irreducible*. Next, let μ_{ij} denote the expected transition time from state i to j, since node behaviors change within finite time, then $\mu_{ij} < \infty$ holds $\forall i, j \in \mathcal{S}$. If let μ_i denote the expected holding time in state i, we have $\mu_i = \sum_{j \in \mathcal{S}} p_{ij}\mu_{ij}$. Thus, $\sum_{i \in \mathcal{S}} \mu_i < \infty$ holds, which implies that $Z(t)$ is also *positive recurrent* by *Theorem 9-2 (pp. 325)* in [12]. Therefore, by *Theorem 9-3 (pp. 327)* in [12], the limiting distribution can be obtained by:

$$P_i \triangleq \lim_{t \to \infty, \forall j \in \mathcal{S}} P(Z(t) = i | Z(0) = j) = \frac{\pi_i \mu_i}{\sum_{j \in \mathcal{S}} \pi_j \mu_j}, \tag{7}$$

where π_i is the stationary probability of state i of X_n.

In order to calculate π_i and μ_i in (7), we must obtain transition probabilities p_{ij} and transition time distributions $F_{ij}(t)$, which are described as follows.

- **Transition Probabilities.**

To determine p_{xi} ($x \in \{C, F, D\}$), we consider both energy consumption and node mobility behavior, which are characterized by an average node lifetime, $\overline{T}_{life}$, and average node residence time, $\overline{T}_{in}$, respectively. To determine p_{xd} ($x \in \{C, S, F\}$), we assume that a destructive node may choose k_a out of total N nodes as victims with probability q_a and needs an average time of $\overline{T}_{eff}$ to compromise these victim nodes. Notice that an inactive node is not affected by actions from destructive nodes, but both faulty nodes and cooperative nodes will be affected. To determine p_{xf} ($x \in \{C, D, I\}$), we assume that destructive and inactive nodes will not become faulty (or more accurately, they will behave like normal nodes, except disruption of network operation). As for cooperative nodes, they are assumed to turn off the packet forwarding function if their residual energies drop below $1/\eta$ of their initial energies, so that they become faulty at time $T_{TS} = \frac{\eta-1}{\eta} \cdot \overline{T}_{life}$. To determine p_{xc} ($x \in \{S, M, F\}$), we assume that a cooperation stimulating mechanism such as *nuglet counter* [31] is used, if the faulty function is due to energy concern. For example, each faulty node possesses a certain number of tokens TC_{max} initially and spends tokens when it sends or receives packets for its own benefit. So faulty nodes must become cooperative if the number of remaining tokens drops below a threshold TC_{thr}. For simplicity, we consider that destructive nodes cannot become cooperative, while it is possible for an inactive node to be repaired or recharged with an average recovery time $\overline{T}_{recr}$. Consider that $\mathbb{P}$ is a stochastic matrix, we can determine p_{xx} ($x \in \mathcal{S}$) correspondingly.

3.1.4 Transition Time Distributions

We use two-parameter *Weibull* distribution from reliability engineering to define $F_{xi}(t)$ as: $F_{xi}(t) = 1 - \exp(-(t/\hat{\beta})^{\hat{\alpha}})$ ($x \in \{C, S, M\}$), where $\hat{\alpha}$ is the *slope* parameter, $\hat{\beta} = \overline{T}_{life}/\Gamma(1 + 1/\hat{\alpha})$ is the *scale* parameter, and $\Gamma(\cdot)$ is the gamma function. $F_{cd}(t)$, $F_{fd}(t)$ and $F_{cf}(t)$ are defined by Weibull distribution similarly. In this work, we assume that $F_{fc}(t)$ is a uniform distribution with the range of $[a, b]$. We further define $F_{ic}(t)$ and $F_{xx}(t)$ ($x \in \mathcal{S}$) by exponential distributions.

$$\mathbb{F}(t) = \begin{pmatrix} \mathcal{E}(\lambda) & \mathcal{W}(\hat{\alpha}, \frac{T_{TS}}{\gamma}) & \mathcal{W}(\hat{\alpha}, \frac{\overline{T}_{eff}}{\gamma}) & \mathcal{W}(\hat{\alpha}, \frac{\overline{T}_{life}}{\gamma}) \\ \mathcal{U}(a, b) & \mathcal{E}(\lambda) & \mathcal{W}(\hat{\alpha}, \frac{\overline{T}_{eff}}{\gamma}) & \mathcal{W}(\hat{\alpha}, \frac{\overline{T}_{life}}{\gamma}) \\ 1 & 1 & \mathcal{E}(\lambda) & \mathcal{W}(\hat{\alpha}, \frac{\overline{T}_{life}}{\gamma}) \\ \mathcal{E}(\frac{1}{\overline{T}_{recr}}) & 1 & 1 & \mathcal{E}(\lambda) \end{pmatrix}, \tag{8}$$

Then the complete definitions of $F_{ij}(t)$ are given by (8), where $\mathcal{W}(\hat{\alpha}, \hat{\beta})$ denotes Weibull distribution with parameter $\hat{\alpha}$ and $\hat{\beta}$, $\mathcal{E}(\lambda)$ denotes exponential distribution with parameter λ, $\mathcal{U}(a, b)$ denotes uniform distribution with range $[a, b]$ and $\gamma = \Gamma(1 + 1/\hat{\alpha})$.

After determining p_{ij} and $F_{ij}(t)$, we obtain π_i by:

$$\vec{\pi} = \vec{\pi}\mathbb{P},\ \sum_{i \in \mathcal{S}} \pi_i = 1,\ \pi_i \geq 0, \tag{9}$$

where $\vec{\pi} \triangleq (\pi_i)$ for $i \in \mathcal{S}$. We further obtain μ_i by:

$$\mu_{ij} = \int_0^\infty t dF_{ij}(t),\ \mu_i = \sum_{j \in \mathcal{S}} p_{ij}\mu_{ij},\ \forall i \in \mathcal{S}. \tag{10}$$

By substituting the results from (9) and (10) into (7), the limiting probability, P_i, can be obtained.

3.1.5 Discussions and Summary

Note that we have the following observations of the proposed nodal model:

(i) *This model is able to characterize the transient and steady behavior of wireless nodes in presence of multiple, interdependent failures and colluding attacks.* The reason for using Semi-Markov Process (rather a continuous-time Markov chain) is because the *sojourn* time during which a node behaves in any state $i \in \mathcal{S}$ may not follow the exponential distribution. For instance, a node is more inclined to be in a failed state due to energy consumption as time passes, and the less residual energy is left, the more likely a node changes its behavior to defective, i.e., not forwarding data for others. The SMP allows for arbitrary distributed sojourn times and can be viewed as a process with an embedded Markov chain (EMC), denoted by $\{X_n\}$, where the state transitions occur at time instants when a node changes its behavior to a *new* state. In other words, this model enables us to understand the evolution of node behaviors over time.

(ii) *This model can be used to describe a wide variety of random threats as explained in the definition of each node state, depending on how to diffuse data into this model.* Specifically, let t_n be the elapsed time between the n-th and $n+1$-th transition, we can define the associated (*time-homogeneous*) *Semi-Markov kernel* $\mathbb{Q} = (Q_{ij}(t))$ by

$$Q_{ij}(t) = Pr(X_{n+1} = j, t_n \leq t | X_n = i) = p_{ij} \cdot F_{ij}(t), \tag{11}$$

where $p_{ij} = \lim_{t\to\infty} Q_{ij}(t) = Pr(X_{n+1} = j | X_n = i)$ is the state transition probability between states i and j, and $F_{ij}(t) = Pr(t_n \leq t | X_{n+1} = j, X_n = i)$ is the transition time distribution from states i to j. The distribution matrix $\mathbb{F}(t) = (F_{ij}(t))$ can be determined by using data diffusion with trace files or by certain probability distribution. To the best of our knowledge, there are no prior works on this problem. Therefore, we plan to use the well-known *Weibull* distribution is used for its wide application in the area of reliability engineering [32, 33, 20]. For instance, the time distribution from cooperative state (C) to inactive (I) $F_{ci}(t)$ is represented by by a two-parameter Weibull distribution $F_{ci}(t) = 1 - \exp(-(t/\hat{\beta})^{\hat{\alpha}})$, where $\hat{\alpha}$ is usually called *slope (or shape)* parameter and $\hat{\beta}$ is usually called *scale* parameter, which can be adjusted by iterative matching with a real-time system [34].

(iii) *This model can be used for complete and incomplete data traces.* One of the concerns for an analytic model is whether it can be used to estimate or predict future behaviors. More importantly, the model itself must be sufficiently generic for data diffusion, given complete or incomplete data.

- When data traces are complete, e.g., peacetime training data is available: With no loss of generality, we can assume that all nodes in the network are cooperative at the initial time, i.e., $Pr(Z(0) = c) = 1$. The transient distributions of the SMP $\{Z(t)\}$, with state space $\mathcal{S}$ and Semi-Markov kernel $\mathbb{Q}$ in (11), satisfy

$$\begin{aligned} P_{ij}(t) &\triangleq Pr(Z(t) = j|Z(0) = i) \\ &= (1 - H_i(t))\delta_{ij} + \sum_{l\in\mathcal{S}} \int_0^t \dot{Q}_{il}(\tau)P_{lj}(t-\tau)d\tau, \end{aligned} \tag{12}$$

  where $H_i(t) \triangleq Pr(t_n < t|X_n = i) = \sum_{j\in\mathcal{S}} Q_{ij}(t)$ is the sojourn time distribution in state i, and δ_{ij} is the Kronecker δ function and defined by 1 for $i = j$ and 0 otherwise. Then the transient distribution $P_{cc}(t)$ is of particular interest, since it indicates the cooperativeness of any node at time $t > 0$. However, it is normally difficult to derive $P_{cc}(t)$ in continuous time domain [35]. Nevertheless, a numerical solution was proposed in [35] to solve (12) by rewriting the transient distributions in discrete-time domain as follows,

$$P_{ij}(mh) = (1 - H_i(mh))\delta_{ij} + \sum_{l\in\mathcal{S}}\sum_{x=1}^{m} h\dot{Q}_{il}(xh)P_{lj}(mh - xh), \tag{13}$$

  where h is the discretization step. In addition, $\dot{Q}_{il}(xh)$ may be further approximated by the difference quotient as,

$$\dot{Q}_{il}(xh) = \frac{1}{h}\left(\hat{Q}_{il}(xh) - \hat{Q}_{il}((x-1)h)\right) \text{ for } x > 1, \tag{14}$$

  where $\hat{Q}_{il}(xh)$ is the empirical distribution of $Q_{il}(\tau)$. By using this method, as long as we have a complete set of trace data that record all state transitions and time instants when transitions occur, the empirical function $\hat{Q}_{il}(xh)$ can be computed by (14), then $P_{ij}(mh)$ can be computed by (13). In deed, this method has already been used in [36] to model behaviors of user mobility based on a large-scale trace database from peacetime[2]. In addition, we can also use peacetime data for training purpose and simulation threats to test the model.

- When data traces are incomplete: Unfortunately, to the best of our knowledge, there is no complete trace data recording user behaviors in wireless ad hoc networks, especially for WMD stressors. Thus, we strive to grasp the stochastic properties of a node malfunction by utilizing any statistics available and reasonable estimations,

  Let T_i be the sojourn time in state i, T_{ij} be the transition time from states i to j, $E[\cdot]$ be the conventional notation for expectation, then we have $E[T_i] = \int_0^\infty (1 - H_i(t))dt$ and $E[T_{ij}] = \int_0^\infty (1 - F_{ij}(t))dt$. In our *preliminary work* [37], we have proved the following:

[2] For example, trace files can be obtained from CERT Statistics at http://www.cert.org/stats/

Lemma 2. *Given the SMP $\{Z(t)\}$ associated with the state space $\mathcal{S}$ and the transition probability matrix (TPM), denoted by $\mathbb{P} = (p_{ij})$, the transient distribution $P_{ij}(t)$ converges to a limiting probability P_j as $t \to \infty$; further, P_j can be calculated by*

$$P_j \triangleq \lim_{t\to\infty} P_{ij}(t) = \frac{\pi_j E[T_j]}{\sum_{l\in\mathcal{S}} \pi_l E[T_l]}, \tag{15}$$

where $\vec{\pi} =< \pi_j >$ is the stationary distribution of $\{X_n\}$.

Note that *Lemma 2* provides a method to estimate the probability of a node being cooperative without a complete set of trace data. It also indicates that statistically, as $t \to \infty$, the probability of a node being in a particular state can be estimated. Moreover, the limiting probability of a node in state i also implies the portion of nodes in state i of a network. For example, a node with 80% probability of being *cooperative* in a network means that there exist 80% nodes in the network are cooperative, which is a clear and concise indication of the network health.

(iv) *This model allows us to determine transition probabilities and mean time of each state* under conditions of distributed, random threats. By following this model, we have obtained mobility-induced failure probability by using our newly developed smooth based mobility model [38, 13]. To identify defective nodes and estimate their mean time to failure as well as failure probability, an energy threshold-based method in which the residual energy of a wireless device can be used to decides whether it forwards data for other nodes or not [39, 40].

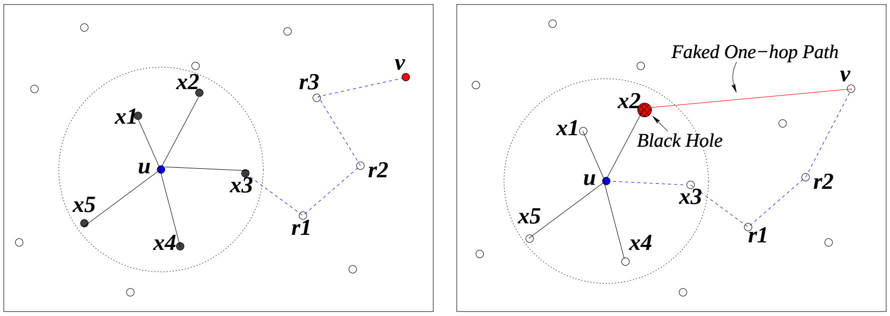
In the following, we will discuss how to use this model to analyze network survivability when multiple failures are present.

3.2 Analysis of Network Connectivity

Recall that our objective is to find out the probability of an ad hoc network keeping k-connectivity in the presence of node failures and communication failures. Based on the proposed node behavior model in Section 3.1.3, we are ready to analyze the connectivity of multi-hop networks stochastically in this section.

3.2.1 Node Isolation due to Misbehavior

We begin our analysis by examining the effects of failures, which is so called *node isolation* problem. Figure 8(a) shows the scenario where all the neighbors of node u are faulty nodes. In this case, the number of node-disjoint *outgoing paths* of u is zero, where the term outgoing path refers the path through which a node can communicate the nodes of at least two-hop away. In the scenario shown in Figure 8(b), one of the neighbors of node u is destructive, e.g., *Black Hole*. In fact, only one *Black Hole* neighbor x_2 is sufficient to trap all traffic initiated from node u if the destination is beyond the neighborhood of node u. In this case, the number of node-disjoint outgoing paths for node u is also zero. If node u is surrounded by one or more other destructive nodes, then the throughput of the data stream via the destructive node will become zero after a short time period, which is especially harmful for long communication sessions.



(a) (b)

Figure 8: Node Isolated by Non-Cooperative Neighborhood.

Let $N_{op}(u)$ denote the number of node-disjoint outgoing paths of node u, then node u is isolated from the network if $N_{op}(u) = 0$. Considering that there exist only two types of destructive nodes in this context, one is to trap all traffic and the other one has abnormal routing functions (e.g., *JellyFish*, we have the following observation:

***Lemma* 3.** *A node u is isolated if it has at least one* Black Hole *neighbor or the total number of faulty,* JellyFish*, and inactive neighbors is d, given it has d neighbors.*

By *Lemma* 3, let D denote the number of neighbors of a node, then we obtain the probability of a node being isolated, given that the node has d neighbors, as

$$Pr(N_{op} = 0|D = d) = 1 - (1 - P_{BH})^d + (1 - P_c - P_{BH})^d, \tag{16}$$

where P_c and P_{BH} are the probabilities that a node is cooperative and a *Black Hole*, respectively. Consequently, a node must have at least one cooperative and no *Black Hole* neighbor to keep it connected to the network.

3.2.2 Condition of Keeping A Node k-Connected

Let $\hat{n}_c(u)$, $\hat{n}_{BH}(u)$ and $\hat{n}_g(u)$ denote the number of cooperative, *Black Hole* and all other neighbors of node u, respectively, then based on the analysis to node isolation problem, we have

***Theorem* 1.** *A node u has k node-disjoint outgoing paths if and only if u has k cooperative neighbors and no* Black Hole *neighbor, i.e., $\{N_{op}(u) = k\} \Leftrightarrow \{\hat{n}_c(u) = k, \hat{n}_{BH}(u) = 0\}$ for $k \geq 1$.*

Notice that the events of any node being in a certain behavior state are mutually independent, then by *multinomial probability law*, we know that the joint distribution of $\hat{n}_c, \hat{n}_{BH}, \hat{n}_g$ is a multinomial distribution. By *Theorem* 1, the probability of a node being k-connected to network, given that the node has d neighbors, is defined as

$$\begin{aligned} Pr(N_{op} = k|D = d) &= Pr(\hat{n}_c = k, \hat{n}_{BH} = 0, \hat{n}_g = d - k) \\ &= \frac{d!}{k!(d-k)!}(P_c)^k \cdot \bar{P}^{d-k},\ k \geq 1, \end{aligned} \tag{17}$$

where $\bar{P} = 1 - P_c - P_{BH}$ denotes the probability of a node being neither cooperative nor *Black Hole*.

3.2.3 Probability of k-Connectivity of Individual Node

Let $\theta(\mathcal{M}_A) = \min\{N_{op}(u) | N_{op}(u) \in \mathbb{N}, u \in \mathcal{M}_A\}$, we have the condition to keep a network k-connected as follows:

***Theorem* 2.** *A multi-hop network $\mathcal{M}_A$ with N_a nodes is k-connected if and only if any active node u of $\mathcal{M}_A$ has at least k node-disjoint outgoing paths, when N_a is sufficiently large.*

Therefore, by *Theorem* 2, the probability of a network being k-connected can be represented by:

$$Pr(\kappa(\mathcal{M}_A) = k) = Pr(\theta(\mathcal{M}_A) \geq k). \tag{18}$$

We assume that the number of outgoing paths for each node u, $N_{op}(u)$, is independent, then from (18), we have:

$$Pr(\kappa(\mathcal{M}_A) = k | N_a) = (1 - Pr(N_{op} < k))^{N_a}, \tag{19}$$

where N_a is the number of active nodes. By the total probability law, we have

$$Pr(N_{op} < k) = \sum_{d=k}^{\infty} Pr(N_{op} < k | D = d) Pr(D = d). \tag{20}$$

To solve this problem, we need to find $Pr(N_{op} < k | D = d)$ and $Pr(D = d)$. By (17), $Pr(N_{op} < k | D = d)$ is given immediately by:

$$\begin{aligned} & Pr(N_{op} < k | D = d) \\ & = 1 - (1 - P_{BH})^d + \sum_{m=0}^{k-1} \frac{d!}{m!(d-m)!} (P_c)^m \cdot \bar{P}^{d-m}. \end{aligned} \tag{21}$$

To derive $Pr(D = d)$, we assume that all nodes move randomly over a finite area with size A. We divide the area into N' small grids virtually so that the grid size is in the same order of the physical size of a node. Consider that the network area is normally much larger than the node physical size, that a node occupies a specific grid, denoted by p', is very small. With large N' and small p', node distribution can be modeled by a *Poisson point process*. Then we have

$$Pr(D = d) \approx \frac{\mu_0^d}{d!} e^{-\mu_0}, \tag{22}$$

where $\mu_0 = \rho \pi r_0^2$. ρ is the node density depending on the underlying mobility model, and r_0 is the transmission range of nodes.

Finally, by (19), (20), (21) and (22), we obtain:

$$\begin{aligned} & Pr(\kappa(\mathcal{M}_A) = k | N_a) \\ & = \left[\frac{\Gamma(k, \mu_0)}{\Gamma(k)} + e^{-\mu_0 P_{BH}} \left(1 - \frac{\Gamma(k, \mu_0(1 - P_{BH}))}{\Gamma(k)} \right) - e^{-\mu_0 P_{BH}} \cdot \frac{\Gamma(k, \mu_0 P_c)}{\Gamma(k)} \right]^{N_a}. \end{aligned} \tag{23}$$

where $\Gamma(\cdot)$ and $\Gamma(h, x) = (h-1)! e^{-x} \sum_{l=0}^{h-1} x^l / l!$ are complete and incomplete Gamma function, respectively.

Recall that the network survivability has been defined in (6) as the probability that all active nodes are k-connected to a network. The survivability of a network $\mathcal{M}$ can be given by the probability that all active nodes have at least k cooperative degree, i.e.,

$$NS_k(\mathcal{M}) \approx Pr(\theta(\mathcal{M}_a) \geq k), \tag{24}$$

where $\mathcal{M}_a$ is the sub-network of $\mathcal{M}$ induced by all active nodes. Based on above equation, we are ready to derive the bounds for network survivability next.

3.2.4 Bounds of Network Survivability

Although (24) offers a guideline on deriving $NS_k(\mathcal{M})$, it is quite challenging to find the distribution of $\theta(\mathcal{M}_a)$. Indeed, $Pr(\theta(\mathcal{M}_a) \geq k)$ is equivalent to the joint probability of every active node being at least k-connected to the network, i.e.,

$$NS_k(\mathcal{M}) \approx Pr\Big(\bigcap_{u \in \mathcal{M}_a} D_c(u) \geq k\Big). \tag{25}$$

We notice that it has been shown that some random graph models do not generate the correlation of the degrees in a pair of adjacent nodes [41]; however, this non-correlation does not imply the independence of node degrees and even cooperative degrees. Considering that deriving the joint probability is actually intractable, we approximate the survivability by finding its asymptotic upper and lower bounds.

To provide an upper bound, recall that our network model described in Section 3.1.3 is a geometric random graph $\mathcal{G}(\mathcal{N}, r)$, in which N vertexes are uniformly and randomly distributed on a 2-D square with area A. The vertex set can actually be represented by a (homogeneous) Poisson point process $\mathcal{H}_\lambda$ with density $\lambda = N/A$. Based on the definition of (homogeneous) Poisson point process, the numbers of points within disjoint subareas are mutually independent random variables (with identical distribution). Thus, we can find $N/(\lambda\pi r^2)$ (active) points, denoted by $\mathcal{N}_D$, so that their transmission ranges (πr^2) are disjoint (non-overlapped) subareas (disks). As a result, the degrees of two nodes u and v are mutually independent as $u, v \in \mathcal{N}_D$. Similarly, $D_c(u)$ and $D_c(v)$ are mutually independent as well. Based on the explanation above, we have an upper bound for $NS_k(\mathcal{M})$ given by

$$\begin{aligned} NS_k(\mathcal{M}) &\leq Pr\Big(\bigcap_{u \in \mathcal{N}_D} D_c(u) \geq k\Big) \\ &= \Big(1 - Pr(D_c(u) < k)\Big)^{\frac{N}{\lambda\pi r^2}}. \end{aligned} \tag{26}$$

Thus, once we obtain the distribution function of cooperative degree, we can calculate the upper bound of survivability.

Next, we explain how to obtain a lower bound for survivability. We first rewrite (25) as

$$NS_k(\mathcal{M}) \approx 1 - Pr\Big(\bigcup_{u \in \mathcal{M}_a} D_c(u) < k\Big). \tag{27}$$

Let N_a denote the number of active nodes in the network, $\mathbf{1}_{\{\mathcal{E}\}}$ denote the indicator function, then we can bound $Pr(\cup_{u\in\mathcal{M}_a} D_c(u) < k)$ from above by using *Boole's inequality*,

$$\begin{aligned} Pr\Big(\bigcup_{u\in\mathcal{M}_a} D_c(u) < k\Big) &= E\left[E\left[\mathbf{1}_{\{\bigcup_{u=1}^{N_a} D_c(u)<k\}}|N_a\right]\right] \\ &\leq E\left[\sum_{u=1}^{N_a} E\left[\mathbf{1}_{\{D_c(u)<k\}}\right]\right] \\ &= E[N_a]\cdot Pr(D_c(u) < k). \end{aligned} \tag{28}$$

Notice that the expected value of N_a is actually equal to $N(1-P_f)$, i.e., $E[N_a] = N(1-P_f)$, where P_i is the (limiting) probability of a node in the failed state, defined in (15). We obtain a lower bound for $NS_k(\mathcal{M})$ as

$$NS_k(\mathcal{M}) \geq 1 - N(1-P_f)\cdot Pr(D_c(u) < k). \tag{29}$$

Again, to solve (29), we need to determine $Pr(D_c < k)$.

From eq (20), we need to find $Pr(D = d)$ and $Pr(D_c < k|D = d)$.

First, to derive $Pr(D = d)$, we use the (de)Poissonization technique presented in [42, 43]. As we mentioned previously, the communication graph of a network $\mathcal{M}$ is associated with a homogeneous Poisson process $\mathcal{H}_\lambda$ with density $\lambda = N/A$. Since we are particularly interested in the topological survivability of active nodes, let $A_0 = \pi r^2$ denote the area covered by a node's transmission range, it is known that the number of active nodes within A_0 is a Poisson random variable with density $A_0\cdot(N_a/A)$. Thus, $Pr(D = d)$ can be approximated by

$$Pr(D = d) = \frac{\mu_a^d}{d!}e^{-\mu_a}, \tag{30}$$

where $\mu = \pi r^2 N(1-P_f)/A$ is the Poisson density. A similar result was also presented in [26], in which more general results were presented for non-uniform node distributions.

Second, we derive $Pr(D_c < k|D = d)$. Since the cooperative degree cannot be greater than the degree for any node, $Pr(D_c < k|D = d)$ is always equal to 1 when $d < k$. When $d \geq k$, $Pr(D_c < k|D = d)$ ($k \geq 1$) can be calculated by

$$\begin{aligned} Pr(D_c < k|D = d) &= \sum_{m=1}^{k-1} Pr(D_c = m|D = d) \\ &+ Pr(D_c = 0|D = d), \end{aligned} \tag{31}$$

in which $Pr(D_c = 0|D = d)$ is the node isolation probability, and $Pr(D_c = m|D = d)$ is the probability of a node being k-connected. With these two items, from (31), we can re-write (31) as

$$\begin{aligned} &Pr(D_c < k|D = d) = 1 - (1-P_B)^d \\ &+ \sum_{m=0}^{k-1}\binom{d}{m} P_c^m\cdot(1-P_c-P_B)^{d-m}. \end{aligned} \tag{32}$$

Thus, by utilizing (30) and (32), $Pr(D_c < k)$ can be obtained from (20), and the upper and lower bounds of the network survivability can be further obtained from (26) and (29). We present our main result next.

3.2.5 Main Result and Implications

Theorem 3. *For a wireless ad hoc network $\mathcal{M}$ in the presence of node misbehavior and failures, when the number of nodes N is sufficiently large, the network survivability defined in* (6) *is upper bounded asymptotically by*

$$NS_k(\mathcal{M}) \leq \left(e^{-\mu_a P_B} \left(1 - \frac{\Gamma(k, \mu_a P_c)}{\Gamma(k)} \right) \right)^{\frac{N}{\lambda \pi r^2}}, \tag{33}$$

and lower bounded asymptotically by

$$NS_k(\mathcal{M}) \geq 1 - N(1 - P_f) \left(1 - e^{-\mu_a P_B} \left(1 - \frac{\Gamma(k, \mu_a P_c)}{\Gamma(k)} \right) \right), \tag{34}$$

where $\mu_a = N(1 - P_f)/(\lambda \pi r^2)$ and λ is the node density, and $\Gamma(h) = (h-1)!$ and $\Gamma(h, x) = (h-1)! e^{-x} \sum_{l=0}^{h-1} x^l / l!$ are the complete and incomplete Gamma functions, respectively.

The above theorem answers and quantifies the impact of different node behaviors on survivability directly. From the upper and lower bounds given in (33) and (33), respectively, we have the following observations by numeric analysis.

1. In general, the survivability is increasing in the cooperative probability P_c, which is accordant with our intuition. When the network area A is fixed, the higher the number of nodes N is, the higher the survivability is, due to the increased density. While if the density is fixed, increasing N will reduce the survivability. This implies that it will become more difficult to achieve the same survivability level as a network scale gets larger without increasing node density substantially.

2. Given two networks $\mathcal{M}_1$ and $\mathcal{M}_2$ with the same N, λ, and P_c, besides cooperative nodes, suppose that $\mathcal{M}_1$ has failed nodes only and $\mathcal{M}_2$ has misbehaving (selfish and *Jellyfish*) nodes only, then $NS_k(\mathcal{M}_1) < NS_k(\mathcal{M}_2)$ always holds. The severer impact of node failures is due to the fact that node failures are also isolated from the network, which reduces the density of active nodes (e.g., μ_a).

3. For given N, P_i, and P_c, both upper and lower bounds of the survivability decreases almost exponentially in $\mu_a P_B$. An interesting observation is that when P_B is not zero, a network with higher density can have a lower survivability. Recall that in Section 3.2.1 we have mentioned that a *Blackhole* node may mislead path selections of its neighborhood and trap surrounding traffics, thus the negative impact of *Blackhole* nodes increases if they are located in the area with high density.

Note that in real networks the nodes at the vicinity of the network (simulation) boundary have less (active) neighbors and thus become isolated easily, which is known as the *border effect*. As pointed out in [44], the border effect is negligible in analysis if the network area is much larger than the transmission coverage area of a single node and the node density is not high. Since the survivability bounds given in (33) and (34) are all asymptotic for sufficiently large N and we are particularly interested in large-scale extended networks (with fixed density) [44], the border effect is not considered in our derivation. For further discussions on the border effect, readers are recommended to refer [26, 44] and the references therein.

Remark 1. *It is a premise that $Pr(D_c < k) < 1/N(1 - P_f)$ should hold to guarantee a positive lower bound given in* (34)*; otherwise, the lower bound is zero. When $Pr(D_c < k) = o(1/N)$, we have the following approximation*

$$\begin{aligned} & 1 - N(1-P_f)\left(1 - e^{-\mu_a P_B}\left(1 - \frac{\Gamma(k, \mu_a P_c)}{\Gamma(k)}\right)\right) \\ \approx & \left(e^{-\mu_a P_B}\left(1 - \frac{\Gamma(k, \mu_a P_c)}{\Gamma(k)}\right)\right)^{N(1-P_f)}, \end{aligned} \tag{35}$$

and the left hand side (LHS) is always less than the RHS in the above equation as $N \cdot Pr(D_c < k) \ll 1$. Since the upper bound given in (33) *is quite loose, we conjecture that the RHS of (35) is a tight upper bound for network survivability. Indeed, if cooperative degrees, $D_c(u)$, are assumed to be independent (as independent degrees assumed in [45]), the RHS of (35) becomes the closed-form approximation for network survivability.*

Remark 2. *A special case of our result in* Theorem 3 *is that all nodes are cooperative and node isolations are due to the lack of neighbors only. In this case, the survivability of a network can be simplified to $(1-\Gamma(k, \lambda\pi r^2)/(k-1)!)^N$ by considering* (34) *and* (35)*, which is the exact probabilistic k-connectivity approximation given in [45]. This indicates that our result provides a more generalized quantitative evaluation on the topological survivability. Moreover, our result, especially the lower bound, is of interest not only for theoretical analysis but also for practical design of survivable wireless ad hoc networks. For example, if the statistics of user behaviors are available, we can use the methods proposed in Section 3.1.3 to estimate state probabilities. Then given a desired survivability preference (e.g., > 0.9), the minimum cooperative degree or the number of nodes can be calculated as theoretical guidance to determine a proper network deployment so that the survivability preference can be achieved with high probability.*

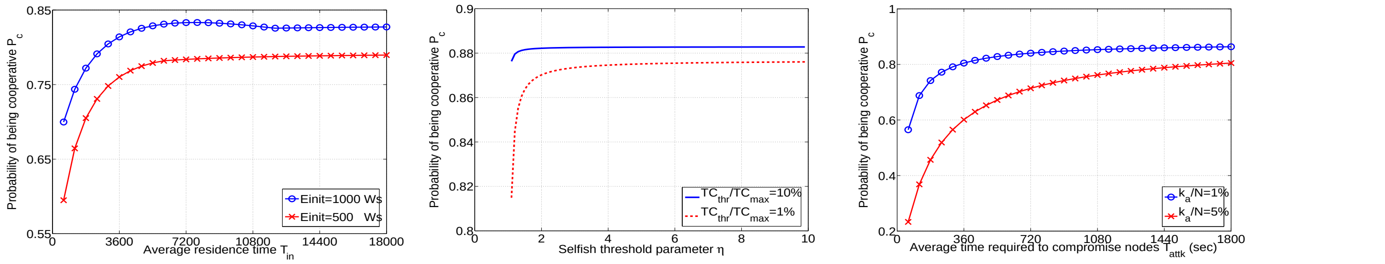
Up to now, we have solved the SNM-Problem by providing the loose upper and tight lower bounds to approximate the network survivability in closed forms, in which the impacts of node misbehavior and failures can be evaluated directly. Next, we conduct exhaustive simulations to confirm our analytical result.

3.2.6 Simulation Results

Up to now, we have obtained the stochastic properties of the impact of node behaviors on network connectivity. In this section, we evaluate our node behavior model and network connectivity of ad hoc networks by simulations.

In this work, we use NS2-v2.27 and MATLAB-v6.5 to perform the simulations. Unless specified otherwise, all simulations are performed in a 1000×1000 m^2 square area, over which 200 mobile nodes with transmission range 150 m are distributed uniformly. IEEE 802.11 is used for medium access control and AODV is used as the routing protocol. *BonnMotion* [46] is used to generate *Gauss-Markov* modeled movement scenarios. In order to calculate the probability of connectivity, we collected the neighborhood statistics of each node per 10 seconds, including the number of neighbors and the behavior of each neighbor. With these information, the number of outgoing paths of each node can be obtained, then the probability of k-connectivity can be calculated.

- **Probability of A Node Being Cooperative**

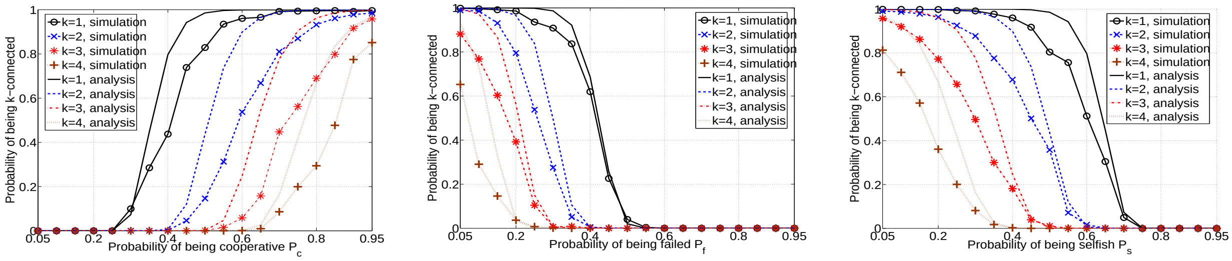


(a) Effect of Node Mobility (b) Effect of Faulty Nodes (c) Effect of Routing Malfunction

Figure 9: Probability of A Node Being Cooperative P_c.

As explained in modeling of node behavior, node mobility is represented by the average residence time $\overline{T}_{in}$. The smaller $\overline{T}_{in}$ is, the faster a node will leave a network, cooperative to its neighbors. As shown in Figure 9(a), the cooperative probability P_c is proportional to $\overline{T}_{in}$ when $\overline{T}_{in} \leq \overline{T}_{life}$ and remains a constant of $1/\overline{T}_{life}$ afterward. From Figure 9(a), P_c is affected by the initial energy E_{init} as well, i.e., a node with a higher E_{init} is more likely to be cooperative. From the discussion of *Stochastic Properties of Node Behavior Model* in Section 3.1.3, as η increases, p_{cs} keeps deceasing until $1/\overline{T}_{life}$, which reflects the fact that a node is more likely to be cooperative if it takes longer time to become selfish. Therefore, in Figure 9(b), P_c increases quickly at the beginning, then almost remains constant afterward. Meanwhile, we can see that a higher token threshold TC_{thr} can increase P_c effectively, which shows that it is necessary to use a cooperation stimulating mechanism to mitigate selfish behavior. Moreover, by Section 3.1.3, the shorter $\overline{T}_{eff}$ is, the more likely a node is compromised to become destructive, which leads P_c in proportion to $\overline{T}_{eff}$, as shown in Figure 9(c). If the fraction of vulnerable nodes within total nodes, k_a/N, is increased from 0.01 to 0.05, then cooperative probability P_c drops dramatically for the same $\overline{T}_{eff}$. Thus, we conclude that external attacks can impact P_c substantially.

- **Probability of k-connectivity** We now study how network connectivity is impacted by misbehaving nodes and node failures. Figure 10(a) shows the simulation results of the probabilities of k-connectivity against P_c for $k = 1, 2, 3, 4$, respectively. In this experiment, P_i and P_{BH} are set to 0 such that we can observe the effect of P_c clearly. From Figure 10(a), the probability of k-connectivity is inversely proportional to k given constant P_c, and proportional to P_c given constant k. To obtain a higher k-connectivity, it is



(a) Effect of Cooperative Probability P_c (b) Effect of Inactive Node Probability P_i (c) Effect of Faulty Probability P_f

Figure 10: Probability of k-connectivity: Effects of Node Behaviors.

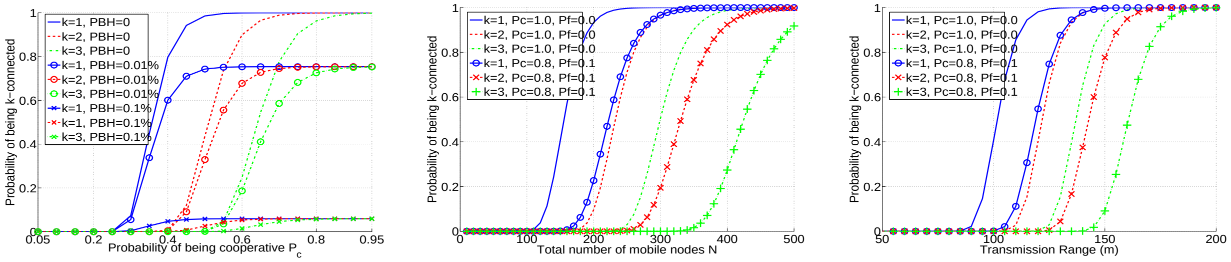
necessary to have a higher P_c.

In order to see the effect of probability of node failure P_i, we set both P_f and P_d as zero to eliminate the impact of misbehaving nodes in our simulations. From Figure 10(b), the probability of k-connectivity decreases very fast as P_i increases. As we expected, for a highly connected network, the impact of P_i is more significant, e.g., the probability of $k = 3$-connectivity drops to 0.4 even as $P_f = 0.2$.

In the same way, we obtain the results from node selfishness as shown in Figure 10(c). Similar to that in Figure 10(b), the plot in this figure indicates that the probability of k-connectivity decreases as selfish probability P_f increases. Nevertheless, differing from the results in Figure 10(b), the probability of k-connectivity does not change significantly when P_f is increased at the beginning, especially for lower k. Notice that the number of active nodes N_a decreases as P_i increases, which makes the network sparser in terms of the decreased node density (e.g., $\rho = N_a/A$ if the node distribution is uniform). Therefore, node failures have severer partitioning effects than selfish nodes.

Compared to the analytical results, the simulation results are lower than analytical ones, which can be explained by the border effect, i.e., the nodes at the vicinity of the simulation boundary have less neighbors and thus become isolate easily. Therefore, the analytical result provides a upper bound for the probability of k-connectivity.

- **k-connectivity Impacted by Other Parameters**



(a) k-Connectivity Impacted by P_{BH} (b) k-Connectivity Impacted by N (c) k-Connectivity Impacted by r_0

Figure 11: Probability of k-Connectivity: Effects of System Parameters.

In addition to node behaviors, we continue to evaluate the impact of other system parameters on network connectivity. Here we look at the effect of *Black Hole* with the probability of P_{BH}. By (23), P_{BH} has tremendous influence on the probability of k-connectivity. Analytical results are illustrated in Figure 11(a) from which we can see that $Black\ Hole$ is the most harmful behavior since it destroys network connectivity much severer than node failures do. Recall the node isolation issue discussed in Section 3.2.1, we find that a *Black Hole* actually can isolate all its neighbors, and its influential scope will be extended when it roams in the network.

Next, we discuss the effect of system size N on the network connectivity. In this simulation, the transmission range r_0 is set as $100m$ to enlarge system size N. Figure 11(b) shows that the required network size N should be enlarged to guarantee the same k-connectivity when destructive or inactive nodes are in present. To discuss the effect of the node's transmission range r_0 on the network connectivity, we change the system size N to 150 from 200 to enlarge the change of r_0. Figure 11(c) shows that the higher k-connectivity is required, the larger r_0 is needed. Similar to the analysis to Figure 11(b), we conclude that the required r_0 has to be increased to guarantee the same k-connectivity if destructive or inactive nodes exist in a mobile ad hoc network.

3.2.7 Summary

In this work, we focused on the modeling and analysis of the impact of node and communication failures to network connectivity of multi-hop wireless networks, which has been rarely studied before. We first classified node behaviors into four types: *cooperative, faulty, destructive* and *failed*, then proposed a node behavior model by employing a semi-Markov process. In our model, mobile nodes change their behaviors according to the well-defined transition probability matrix and transition time distribution matrix. After obtaining the limiting probability of a node being in each behavior state. we analyzed the node isolation problem resulting from misbehaving neighbor nodes and provided the condition under which mobile nodes can be connected with a mobile ad hoc network. In consequence, we obtained the close-form and upper bound of network survivability, that is, the probability of a network being k-connected.

3.3 k-Connectivity Routing

In this project, we target military networking infrastructures such as combat strategic systems, tactical ad hoc networks, which carry time-sensitive information and demand for reliable and non-disruptive communications. after studying the analysis of network survivability in the presence of multiple failures as a consequence of node mobility, energy depletion and operations, and various nodes faulty behaviors. We also derived network survivability when communication operations are interrupted. Through detailed analysis, we obtain a close-form representation of probability of node isolation and survivability for k -connected networks. Based on our models and analysis, we aim to to design a network protocol that is robust against random failures and routing misbehavior.

3.3.1 Objectives and Approaches

How to improve the network performance or at least maintain a graceful performance degradation of wireless multi-hop networks in the presence of misbehaving nodes is an important design issue and research problem. Previous works on tackling misbehaving nodes in wireless multi-hop networks can be classified into three categories: cryptographic-based secure routing protocols, incentive- based cooperation stimulating mechanisms, and multipath-based reliable forwarding schemes. However, all existing works cannot fully mitigate the impact of faulty nodes which may be routing-compliant or may not be malicious nodes. Instead, they cannot operate property due to random threats, such as dysfunctional devices, connection or link failures, incomplete knowledge of function and trustworthiness of other nodes caused by disruption as a result of biological, chemical, electromagnetic pulses, or dysfunctional devices; and faulty networking behaviors due to cyber-attacks. More importantly, none of previous reputation schemes has ever considered whether excluding faulty nodes will impair the network connectivity, which is, however, a prerequisite for all communications. Therefore, the routing-compliant misbehavior is an open and challenging problem for the design of wireless multi-hop networks.

We *approach* the k-connectivity routing from the perspective of topology control because the DTRA mission is to provide strategies of robust network architecture. More specifically, we strive to design a resilient topology over a network such that network operations on both control and data planes are distributed only between cooperative neighbors and the network is k-connected with high probability (e.g.,≥ 0.9). We first define a new metric called resilient capacity to measure the maximum number of destructive nodes that a network can sustain under the constraint that the network is k-connected with a certain probability. By analyzing the theoretical bound on resilient capacity, we are able to know how many destructive nodes can be excluded without impairing the connection of the network (to a certain level). Further, we show that an optimal resilient topology can maximize the resilient capacity and satisfy the connectivity requirement if all and only cooperative nodes are included in the topology in which each node has at least k-cooperative neighbors and the average degree scales with $\log(N)$, where N is the network size.

To achieve the optimal topology, we next determine node behavior dynamics by proposing a node cooperativity measurement scheme to quantify the likelihood of any node being cooperative. By using the measurement scheme, we then design a distributed topology control protocol called PROACtive to enhance the network resilience to routing-compliant misbehavior. By applying the PROACtive protocol, every node is able to select more than k cooperative neighbors and exclude misbehaving nodes from its neighbor set dynamically. As a result, the union of all cooperative neighbor sets forms a resilient topology which satisfies the connectivity requirement and maximizes the resilient capacity (at the best effort).

Compared with other works, the design of PROACtive protocol has several unique features and advantages. First, the reliable data delivery is achievable with the assistance of the routing protocol since control packets are only dispersed among the member nodes in the generated topology; second, our approach does not involve new security vulnerabilities and can avoid the false accusation problem; third, our protocol is quite light-weight in the computation and communication complexity and has a bounded convergence time. Finally, by implementing the PROACtive protocol in the network simulator tool ns2, we confirm that after applying routing protocols (e.g., AODV) on the topology generated by the PROACtive protocol, the network goodput can be improved significantly in different network scenarios. Further, the protocol is quite scalable due to its low overhead and fast convergence, and performs even good in networks with high node mobility. Therefore, the PROACtive protocol is a very promising and feasible solution for wireless multi-hop networks to enhance their resilience to more sophisticated node misbehavior.

3.3.2 Resilience Capacity and PROActive Protocol

Our contributions include the rigorous definition and analysis of *resilience capacity*, *necessary conditions for probabilistic k-connectivity*, and *design of PROActive protocol.*

To define the *resilient capacity* of a network rigorously, let $(\Omega, \mathscr{F}, Pr)$ be the probability space on which the random link connection of mobile nodes is defined. In particular, Ω is the sample space consisting of all the possible topology G of a network, $\mathscr{F}$ is a σ-field in Ω and Pr is a probability measure on $\mathscr{F}$. From graph theory, we know that the connectivity of a graph G, denoted by $\kappa(G)$, is the maximum k such that G is k-connected [25]. Due to node mobility and random node behavior, the topology of a given wireless multi-hop network is dynamically changing at all times, which results in the variant connectivity. Thus, the connectivity of a dynamic network can be treated as a random variable defined on Ω and the probabilistic k-connectivity of a network can be defined by $Pr(\{G \in \Omega : \kappa(G) = k\})$, or simply $Pr(\kappa(G) = k)$, for G as the geometric random graph model of the network. With above notations, we define the resilient capacity as below

Definition 2. *Given a wireless multi-hop network, let its topology be represented by a geometric random graph $G_{\mathcal{N},r}(N_m^0)$, where N_m^0 is the initial number of destructive nodes in the graph. For a connectivity preference $0 < \psi_0 < 1$, the resilient capacity of the (topology) graph is defined by*

$$\Lambda(\psi_0, G) \triangleq \max\{N_m^* - N_m^0, 0\} \tag{36}$$

where $N_m^* = \max\{N_m : Pr(\kappa(G_{\mathcal{N},r}(N_m)) = k) \geq \psi_0\}$.

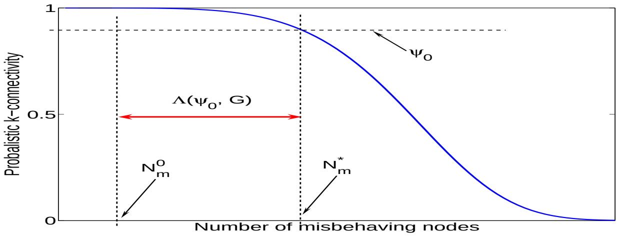


Figure 12: The relation between the resilience capacity and probabilistic k-connectivity.

The physical meaning of the resilient capacity is the maximum number of *extra* destructive nodes that a topology can accommodate such that the topology is still k-connected with a certain probability, as illustrated in Figure 12. If $N_m^* \leq N_m^0$, we define $\Lambda(\psi_0, G)$ as 0, implying that no cooperative node can become destructive any more without degrading the probabilistic k-connectivity below a certain level. In fact, the resilient capacity also measure the maximum number of extra misbehaving nodes that can be excluded from the topology.

However, it is worthy of noting that whether a node can establish reliable connections to other nodes depends on whether the node has *cooperative* adjacent nodes that operate normally on both control and data planes. Let $D_c(\omega)$ be the number of cooperative adjacent nodes of node ω, called the *cooperative degree* of ω, we define $\theta(G_{\mathcal{N},r}(N_m))$ (or simply $\theta(G)$) as the *minimum cooperative degree* of a graph $G_{\mathcal{N},r}(N_m)$, i.e., $\theta(G) \triangleq \min\{D_c(\omega), \forall \omega \in G\}$. As a result, we have

***Proposition* 1.** *For a wireless multi-hop network represented by $G_{\mathcal{N},r}(N_m)$, let μ be the average number of nodes in a node's transmission range. Suppose $N \gg 1$, then for any positive integer $k \geq 1$,*

$$\begin{aligned} Pr(\kappa(G) = k) &\approx Pr(\theta(G) \geq k) \\ &\geq 1 - N\left(\frac{\Gamma(k, \mu(1 - P_m))}{\Gamma(k)}\right), \end{aligned} \tag{37}$$

where $\Gamma(h) = (h-1)!$ and $\Gamma(h, x) = (h-1)!e^{-x}\sum_{i=0}^{h-1} x^i/i!$ are the complete and incomplete Gamma functions, respectively.

Remark 3. Proposition 1 *implies that the necessary condition for a network to be k-connected is that every node should have at least k cooperative adjacent nodes. Thus,* (37) *provides us a useful tool to design a k-connected topology w.h.p. in a localized and distributed algorithm. In addition, from* (37)*, we know that $Pr(\kappa(G) = k)$ is a decreasing function in P_m, which implies that the more misbehaving nodes a network has, the harder for the network to keep its topology k-connected w.h.p.. By using this result, we are able to determine the resilient capacity, presented next.*

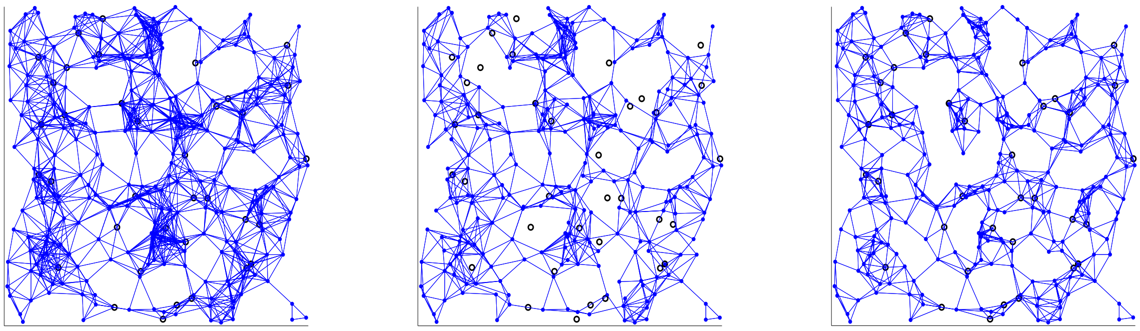
Proposition 2. *Given a network modeled by $G_{\mathcal{N},r}(N_m^0)$, let $\mathcal{N}_c$ be the set of cooperative nodes in G with $N_c = |\mathcal{N}_c|$. Let the topology containing all cooperative nodes be denoted by $G^-_{\mathcal{N}_c,r}(0)$, and let $G'_{\mathcal{N}',r}(N'_m)$ denote any topology containing both cooperative and misbehaving nodes for $0 < N'_m < N_m^0$ and $|\mathcal{N}'| = N_c + N'_m$. Then $\Lambda(\psi_0, G^-) > \Lambda(\psi_0, G')$ holds for any $0 < \psi_0 < 1$ and $k = 1$.*

Remark 4. *The result of* Proposition 2 *implies that the resilient capacity can be maximized when the generated topology contains* only *and* all *cooperative nodes of the original network. The above analysis also provides a new insight on the trade-off between eliminating destructive and tolerating destructive in that the more existing destructive nodes can be deleted from the topology, the more new constructive nodes can be accommodated. Nevertheless, under the constraint of connectivity, the maximum number of sustainable misbehaving nodes is upper bounded such that not all existing misbehaving nodes can be eliminated if the resilient capacity is already zero.*

With above understanding, it is clear that *theoretically* it is possible to design an optimal routing protocol to achieve the maximum resilience capacity under connectivity constraints. The details of our protocol will not be discussed here. Instead, we highlight our results and observation here. To evaluate the performance of our solution, we implement the PROACtive protocol in the simulation tool $ns2$ and make three modifications to the existing AODV module. First, the promiscuous mode is supported such that every node can measure others' cooperativities; second, $RREQ$ and $RREP$ messages are distributed only among neighbors such that path selections are controlled within the topology generated; third, the destructive effect is introduced by letting nodes configurable to drop data packets to be forwarded randomly.

Figure 13(a) shows a network without applying any topology control. Figure 13(b) shows the topology generated by the PROACtive protocol, in which cooperative and destructive nodes are represented by solid dots and circles, respectively. From the figure, we can see that the topology excludes most of destructive nodes, while keeping most of cooperative nodes connected. To highlight the difference from other topology control protocols, the topology generated by the *K-Neigh* protocol (Phase 1 only, with $K = 9$) [47] is shown in Figure 13(c). It is no wonder that all misbehaving nodes are included in Figure 13(c) because the neighbor selection in K-Neigh is only based on the distance between nodes.

To test connectivity, we use the DFS (depth-first-search) algorithm to calculate the maximum number of nodes can be removed without partitioning the network. Then the probabilistic k-connectivity is calculated by the ratio between the number of k-connected topologies and that of all topologies tested. In Figure 14(a), we can see that the k-connectivity probabilities of generated topologies keep beyond 0.9 when $N > 700$ for both $P_m = 10\%$ and $P_m = 40\%$ when original networks are almost physically k-connected. Nevertheless, when $N < 500$, the k-connectivity probabilities for generated topologies and original networks decrease dramatically even down to 0.



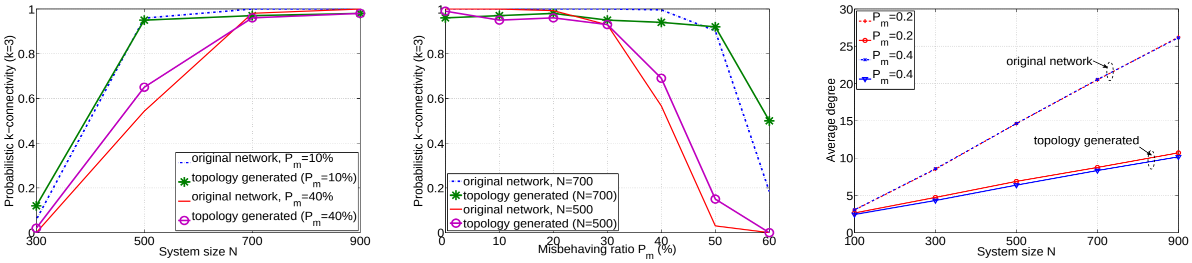
(a) No topology control

(b) Topology generated by PROACtive

(c) *K-Neigh* with $K = 9$ (Phase I)

Figure 13: The illustration of the topology generated, compared with the original network (circles: misbehaving nodes, dots: cooperative nodes).

Further, we observe that the k-connectivity can hardly be preserved if P_m is too high, e.g., $P_m > 30\%$, as shown in Figure 14(b). Another observation is that the average (node) degree is reduced considerably in the generated topologies, as shown in Figure 14(c), and it is decreasing slightly in the misbehaving ratio P_m because of the less chance to find enough cooperative neighbors. Recall that $\mu = \Theta(\log N)$ is a condition for connectivity, from Figure 14(c) we can see that the average degree of generated topologies is asymptotically greater than $\log N$ (given the original networks k-connected), which satisfies the second condition of the connectivity constraint and also implies the effectiveness of our neighbor cooperativeness threshold.



(a) k-connectivity vs. N

(b) k-connectivity vs. P_m

(c) Average degree vs. N

Figure 14: The preservation of k-connectivity in the generated topology.

3.3.3 Summary

The main objective of this study is to find recovery strategies, or more importantly, to *proactively* deliver data in a timely manner. Therefore, our definition of resilience capacity is the corner-stone toward such an objective because we need a rigorously defined metric for optimal design. Our results have demonstrated following features:

- *Localized and distributed algorithm*: This protocol is fully distributed and it requires local topology formation only.
- *Preservation of k-connectivity*: This protocol can preserve the connectivity of generated topology w.h.p. (≥ 0.9) if the underlying network is physically k-connected.
- *Acceptable false positive (negative) ratio*: This protocol can avoid the case of the exclusion of cooperative nodes, called *false positive*, and the case of the inclusion of destructive nodes, called *false negative*.
- *Low overhead and fast convergence*: This protocol is light-weight in the computation and communication complexity and have a bounded convergence time.
- *Interoperability with routing protocols*: This protocol is interoperable with routing protocols to provide a graceful performance degradation.

3.4 Network Devolution Under Random Failures

3.4.1 Objectives and Approaches

A prerequisite for any communications in multi-hop networks is that the underlying topology should be connected. Although many works have been done to provide theoretical guidance to achieve an asymptotic full connectivity, this full connectivity is impractical to achieve all the time when wireless multi-hop networks scale larger and larger. Thus, we find the fraction of nodes in the largest connected component (giant component) to be a better metric to evaluate the resilience of a large network to failures, that is, the larger the giant component is the more resilient a network shall be. Consequently, understanding the network devolution process when the number of node failures increases, especially the critical time when the network experiences topological transitions, is of importance in both theory and practice.

In this subtopic, we focus on the following question: *for a large-scale wireless multi-hop network in the presence of random failures, when does the network change its behavior from an almost connected phase to a fully partitioned phase?* Here a network is said to be almost-connected if there exists a giant component that is composed most of surviving nodes with high probability; while a network is fully-partitioned if no such a giant component exists asymptotically almost surely. *To tackle this problem*, we couple a network devolution process with a continuum percolation process in a geometric random graph with uniform node distribution. By using the concept of percolation probability, we first define two metrics, the *last connection time* and *first partition time,* to characterize the critical phase transition time. The former is the last time at which a network is almost connected and the latter is the first time at which a network is fully partitioned. Then we analyze the conditions under which a geometric random graph does (not) have a giant component of surviving nodes.

Our approach takes following procedures. We first map the percolation process defined on the continuous plane onto a discrete lattice, whose edges are declared *open* if certain properties of the Poisson process in their vicinity are met (*closed* otherwise). In the discrete lattice, we then investigate the condition when infinite open paths (composed of open edges) exist with positive probability. With a careful definition on the open edge in the lattice, a reverse mapping can be carried out back to the continuous plane so that infinite open paths on the discrete plane indicate connected components on the continuous plane. Finally, we obtain the continuum percolation conditions, which enable us to derive the bounds of critical phase transition time, i.e., $t_c(n)$ and

$t_p(n)$, for given survival functions.

Main Results: To the best of our knowledge, this is one of the first studies on network evolution problem. Therefore, our definitions and formulation of problems have laid a good foundation for future research.

In order to understand the critical phase transition time during network devolution, we first define *almost connected* and *fully partitioned* networks as follows

Definition 3. *Let $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$ be a geometric random graph, in which every point is associated with the same survival function $S(t)$. Let $\lambda_1(t) \triangleq \lambda_0 S(t)$, then the network represented by $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$ is said to be almost connected if $p_\infty(\lambda_1(t)) > 0$, and fully partitioned if $p_\infty(\lambda_1(t)) = 0$, where $p_\infty(\cdot)$ is the percolation probability.*

Next we define two new metrics called the *last connection time* and *first partition time*.

Definition 4. *With the same conditions given in* Definition 3*, the* last connection time *is defined by*

$$t_c(n) \triangleq \sup\{t > 0 : p_\infty(\lambda_1(t)) > 0\}, \tag{38}$$

where $\lambda_1(t) \triangleq \lambda_0 S(t)$. The first partition time *is defined by*

$$t_p(n) \triangleq \inf\{t > 0 : p_\infty(\lambda_1(t)) = 0\}. \tag{39}$$

Definition 5. *The critical phase transition time, denoted by $\mathcal{T}_C$, is the critical time point above which $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$ is disconnected a.a.s. (sub-critical) and below which $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$ is connected a.a.s., (supercritical), that is*

$$\lim_{n\to\infty} Pr(G \textit{ is connected}) = \begin{cases} 1, & \textit{if } t < \mathcal{T}_C, \\ 0, & \textit{if } t > \mathcal{T}_C. \end{cases} \tag{40}$$

The exact value of T_c is unknown, but it is expected to be bounded by $t_c(n)$ and $t_p(n)$ from below and above, respectively, based on our definitions on $t_c(n)$ and $t_p(n)$.

3.4.2 Network Partion Problem and Theoretical Limits

Now we formulate the problem addressed in this paper as the *Network Partition Time (NPT)* problem.

Definition 6. (NPT problem)*: For a large-scale wireless multi-hop network represented by a geometric random graph $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$, every node is assumed to be independently associated with a common survival function $S(t)$. Given the network is fully connected at initial time $t = 0$, find out*

1. *the relations among n, λ_0, r_n, and $S(t)$ that would be sufficient to guarantee that the network is almost connected or fully partitioned, respectively;*

2. *the upper limit of $t_c(n)$ and the lower limit of $t_p(n)$, such that the critical phase transition time $\mathcal{T}_C$ can be bounded by these limits.*

The results of this problem reveal that when time $t < t_c(n)$, the network is guaranteed to be almost connected (super-critical); while $t > t_p(n)$ will be sufficient for the network to be fully partitioned (sub-critical). We expect the bounds of critical phase transition time (i.e., $t_c(n)$, $t_p(n)$) to be tight so that the phase transition is sharp and the *period of phase transition* (i.e., the gap between $t_c(n)$ and $t_p(n)$) converges to 0 as fast as possible. Nevertheless, in practical, a longer period of phase transition might be preferable to provide a smooth degradation of connectivity.

Our theoretical results can be summarized as follows:

Theorem 4. *Given a graph $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$, assume $\mu_0 = \lambda_0 \pi r_n^2 = \Theta(\log n)$ and the degree bound $K = (1 + \epsilon_n)\mu_0$ where ϵ_n is an arbitrary increasing function of n. There exists a positive constant c_ϵ, such that if the survival function $S(t)$ satisfies,*

$$S(t) > \frac{\sqrt{5}(\ln 18 - \ln(1 - 15\phi_n))}{c_\epsilon r_n \sqrt{\lambda_0 \ln n}}, \tag{41}$$

where $\phi_n = 1 - \exp(-\frac{2c_\epsilon^2 \ln n}{\epsilon_n^2 \mu_0})$, then $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$ is in the super-critical phase.

Theorem 5. *Given a graph $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$, assume $\mu_0 = \Theta(\log n)$ and $K = (1 + \epsilon_n)\mu_0$. There exists a positive constant c_ϵ, such that if the survival function $S(t)$ satisfies:*

$$S(t) < \frac{\ln \sqrt{3\psi_K} - \ln(\sqrt{3\psi_K} - 1)}{c_\epsilon r_n \sqrt{\lambda_0 \ln n}}, \tag{42}$$

where $\psi_K = \frac{\Gamma(K,\mu_0)}{(K-1)!}$ and $\Gamma(x, y)$ is the incomplete Gamma function, then $G(\mathcal{H}_{\lambda_0,s_n}, r_n)$ is in the sub-critical phase.

Remark 5. The assumptions on μ_0 and K are needed to achieve an initial fully connectivity, which is the condition of the NPT problem. Further, they also guarantee $\psi_K > \frac{1}{3}$ such that $\ln(\sqrt{3\psi_K} - 1)$ is a real number. In fact, the format of K indicates that the impact of interference has to be sufficiently small so that the degree bound could be large enough to support the percolation as $n \to \infty$, which is accordant with the result proved in [48].

Next, the following corollaries answer the second part of the NPT problem, providing the theoretical bounds on the critical phase transition time.

Corollary 1. (Limits of $t_c(n)$ and $t_p(n)$ with light-tailed $S(t)$): *Assume the survival function $S(t) = e^{-\alpha t}$ (exponential), where the positive $1/\alpha$ represents the mean lifetime of a node, then the upper limit of last connection time $t_c(n)$ is,*

$$t_c(n) = \frac{1}{\alpha} \ln(\ln n) + c_1 \ \sim \Theta(\log(\log n)), \tag{43}$$

where $c_1 = \frac{1}{\alpha}(\ln(c_\epsilon \sqrt{\frac{c}{\pi}}) - \ln(\sqrt{5} \ln \frac{18}{1-15\phi_n}))$ and $c \triangleq \frac{\mu_0}{\ln n}$. The lower limit of first partition time $t_p(n)$ is,

$$t_p(n) = \frac{1}{\alpha} \ln(\ln n) + c_2 \ \sim \Theta(\log(\log n)), \tag{44}$$

where $c_2 = \frac{1}{\alpha}(\ln(c_\epsilon \sqrt{\frac{c}{\pi}}) - \ln(\ln \frac{\sqrt{3\psi_K}}{\sqrt{3\psi_K - 1}}))$.

Corollary 2. (Limits of $t_c(n)$ and $t_p(n)$ with heavy-tailed $S(t)$): *Assume $S(t) = (t/\eta)^{-\rho}$ (heavy-tailed Pareto, $\rho > 1$) with mean $\frac{\eta\rho}{\rho-1}$, then the upper limit of $t_c(n)$ is,*

$$t_c(n) = c_3(\ln n)^{1/\rho} \sim \Theta((\log n)^{1/\rho}), \tag{45}$$

where $c_3 = \eta(\frac{c_\epsilon\sqrt{c/\pi}}{\sqrt{5}(\ln 18-\ln(1-15\phi_n))})^{1/\rho}$ and $c \triangleq \frac{\mu_0}{\ln n}$. The lower limit of $t_p(n)$ is

$$t_p(n) = c_4(\ln n)^{1/\rho} \sim \Theta((\log n)^{1/\rho}), \tag{46}$$

where $c_4 = \eta(\frac{c_\epsilon\sqrt{c/\pi}}{\ln(\sqrt{3}\psi_K)-\ln(\sqrt{3}\psi_K-1)})^{1/\rho}$.

Remark 6. A premise used in above theorems is $\mu_0 = c\ln n = \Theta(\log n)$ where c is some constant such that the network is fully connected initially. In particular, Xue and Kumar proved in [49] that $5.1774\log n$ is required for a.a.s. connectivity and this threshold was further improved by Balister et al in [50] to $0.5139\log n$ (also see [51]). However, in our simulations, we find that $0.5139\log n$ is far less sufficient to achieve an initial connected random topology and actually $5.1774\log n$ is a "good" threshold for connectivity.

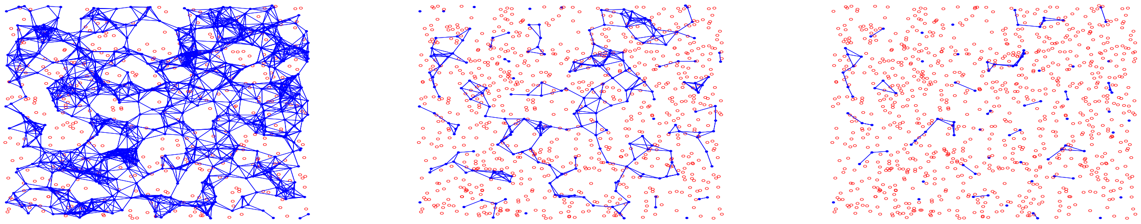
Remark 7. From reliability engineering, we know that many lifetime distributions (e.g., exponential, log-normal, Pareto, Weibull) are either light-tailed or heavy-tailed according to the decay speed of their tails. Since the exponential distribution is the only distribution to have a constant failure rate and applies naturally to model memoryless lifetime, it is used to represent light-tailed survival functions; while the Pareto distribution is used to represent heavy-tailed survival functions when node lifetime is power law or with very large variance.

Remark 8. A network with Gaussian node distribution is more resilient to random failures, in terms of a graceful degradation on the relative giant component size and a longer network survival time, compared with the counterpart with uniform node distribution. This finding implies that the node distribution can be also an important factor in determining the overall network resilience to random failures.

3.4.3 Simulation Results: Visible Network Devolution

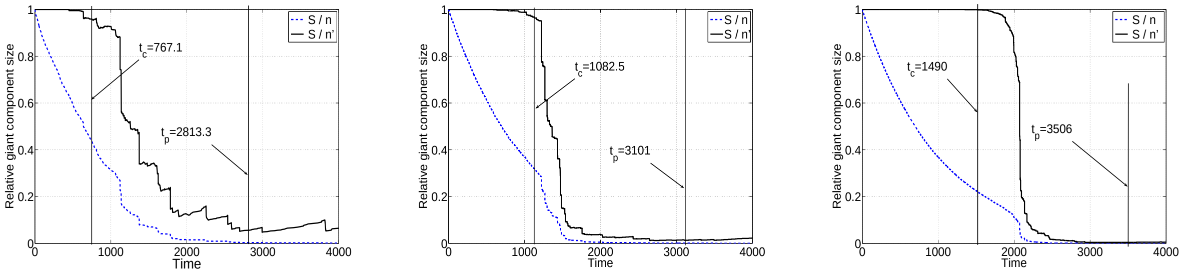
To emulate the devolution process, nodes fail one by one in the increasing sequence of their lifetimes. Upon node failures, we use a depth-first search (DFS) algorithm to record all components induced by surviving nodes and calculate the *giant component size* S (i.e., the number of surviving nodes in the largest component). The *relative* giant component size is defined by $S_R \triangleq S/n'$ where n' is the number of remaining surviving nodes, in order to characterize the phase transition phenomenon.

Figure 15 illustrates an example of the topological devolution process of a graph of 1000 nodes, where solid dots and circles represent surviving nodes and failed nodes, respectively. The survival function is Pareto with parameters set above. By using (45) and (46) and choosing $c_\epsilon = 2.5$, we have $t_c(n) = 733.8$ and $t_p(n) = 2041$. As expected, When $t < t_c(n)$, the topology constructed by remaining nodes is almost connected with a single giant component, as shown in Figure 15(a). On the contrary, when $t > t_p(n)$, the network is fully partitioned and has only several small components, shown in Figure 15(c). Figure 15(b) shows the topology in the period of phase transition, i.e., $t_c(n) < t < t_p(n)$, where the network are disconnected into parts but with a component larger than others.



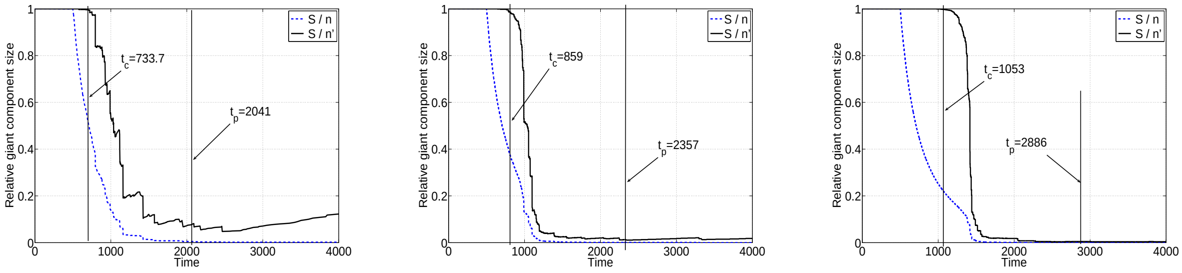
(a) $t = 729.5 < t_c(n)$ (b) $t_c(n) < t = 1204.4 < t_p(n)$ (c) $t = 2193.4 > t_p(n)$

Figure 15: Snapshots of the graph devolution process at different times.



(a) $n = 1k, r_n = 138$ (b) $n = 10k, r_n = 178$ (c) $n = 10k, r_n = 195$

Figure 16: The phase transition and critical time bounds with exponential survival functions ($\alpha = 0.001$).



(a) $n = 1k, r_n = 138$ (b) $n = 10k, r_n = 178$ (c) $n = 100k, r_n = 195$

Figure 17: The phase transition and critical time bounds with Pareto survival functions ($\rho = 2.0, \eta = 500.0$).

Figures 16 and 17 show clearly how the relative giant component size ($S_R \triangleq S/n'$) decreases when the network experiences increasing random failures. We summarize our observations as follows. First, the period of phase transition is bounded by the theoretical limits of $t_c(n)$ and $t_p(n)$ in all simulation scenarios, which confirms the correctness of our analytical results. Second, as expected, the larger the network size n is, the sharper the phase transition is, which is true for both light-tailed and heavy-tailed survival functions. Third, compared with the actual value of the giant component size S (or the ratio S/n), it is clear that the relative giant component size, i.e., $S_R = S/n'$, is a more appropriate metric to indicate the phase transition phenomenon in the devolution process due to random failures. Finally, a surprising observation is that given the same n and average node lifetime, the network with Pareto survival function decomposes substantially faster than the network with exponential survival function. To explain this phenomenon, it is noticed that the variance of Pareto-distributed lifetimes is much larger than that of exponential-distributed lifetimes. This implies that the majority of Pareto-distributed lifetimes have to be short enough to compensate a small number of huge lifetimes, in order to achieve the same average lifetime with exponential-distributed lifetimes. Consequently, more nodes with Pareto-distributed lifetimes fail earlier than the nodes with exponential-distributed lifetimes.

4 Correlated failures and their propagation in inhomogeneous networks

4.1 Failure Propagation via Multi-Hop Communications

During the course of studying fundamental limits of network responses to attacks, we found our understanding of network architecture is very limited for which we hardly find any literature on the topic. For instance, once a failure is detected or occurred, how fast can such failures be known (by one or more nodes) in the network? The problem is important in two-fold: first it is critical the design of recovery strategies because a network should be designed sufficiently robust against the impact of such failure; second, the network (and protocols) needs to be designed intelligently such that the information of failure can be made available to as many nodes as possible within the shortest time period. Therefore, a fundamental problem is: what is the speed of information propagation?

4.1.1 Objectives and Approaches

In the pioneering paper [52], Zheng shows that there is a constant upper bound W on the information diffusion rate and the network is able to achieve a constant diffusion rate, regardless of the network population, in both the *extended* and the *dense* networks. Achieving W requires three conditions: i) every node uses an optimal transmission radius R, ii) the transportation distance of a packet is a multiple of R, and iii) the relay nodes are aligned with separation distance R. Lacking any of these conditions results in W unreachable.

However, a few interesting questions remain unanswered yet. First, if the packet transportation distance is known and not equal to a multiple of R, what is the best propagation strategy for the packet to achieve the fastest delivery? Since W is unreachable, is there a tighter speed upper bound? Second, when delivering a packet, we care about both *how fast* and *how well* the packet is delivered, that is, whether all the intended recipients can receive the packet successfully. When the satisfaction of packet delivery without missing any recipient is considered, what is the speed upper bound under this constraint? Third, if the optimal transmission radius R is used but the relay nodes are not perfectly located, what is the gap between the actually achieved speed and the desired upper bound W? We attempt to provide the answers to these three questions in this work.

Main Results: Our objective is to find the speed of propagation in an arbitrary network which nodes are *not* placed evenly and they have different transmission ranges.

Before we investigate the speed of information propagation, it is necessary to understand how information propagates in multihop wireless networks. An illustration is shown in Figure 18, in which a packet is originated by node v_0 at time zero. Node v_0 chooses a transmission radius r_{v_0} and sends out the packet. The packet is

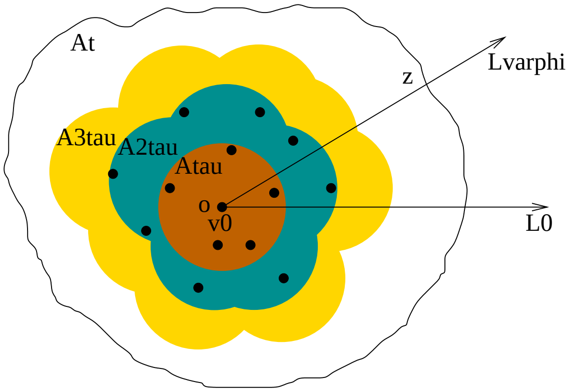


Figure 18: Information propagation in multihop networks.

received by all the relay nodes in $\mathcal{A}_{v_0}$ by the end of v_0's transmission. These relay nodes then forward the packet by rebroadcasting it. Figure 18(a) depicts the area reached by the packet after two hops.

Denote $\mathcal{V}(t)$ as the set of nodes that have received the packet by time t and $\widetilde{\mathcal{V}}(t) \subset \mathcal{V}(t)$ as the subset that have forwarded the packet. The total area that the packet has reached by time t is expressed as $\mathcal{A}(t) = \cup_{v_i \in \widetilde{\mathcal{V}}(t)} \mathcal{A}_{v_i}$. In addition, let $\mathcal{L}_\varphi$ denote the line starting from v_0 toward the direction $\varphi \in [0, 2\pi)$ and $\mathcal{L}_\varphi(t) = \mathcal{L}_\varphi \cap \mathcal{A}(t)$. In Figure 18(b), $\mathcal{L}_\varphi(t)$ is the line segment $\overline{oz}$. The *Information Propagation Speed* in the direction φ is then defined to be

$$w_\varphi(t) = \frac{|\mathcal{L}_\varphi(t)|}{t}, \tag{47}$$

where $|\mathcal{L}_\varphi(t)|$ is the length of $\mathcal{L}_\varphi(t)$.

As the first contribution, we show that there is another optimal transmission radius other than R if the packet transportation distance is not a multiple of R. We note that in broadcast communications, as the locations of packet recipients may not be known in advance, R is the best transmission strategy. In unicast communications, however, the location of the packet recipient may be known. If the known transportation distance is not a multiple of R, another transmission radius that optimally fits the specific distance should be used. Interestingly, we find that there is a unified optimal transmission radius and speed upper bound in large wireless networks.

As the second contribution, we determine the speed upper bound under the constraint of guaranteeing a given level of packet delivery satisfaction. We examine two different noise models. In the first model, the noise in the network is determined by the background Gaussian noise. We show that there exists a threshold node density, above which there is a constant speed upper bound. In the second model, interference is the determinant source of noise. We show that there also exists a threshold node density, above which, however, the speed upper bound decreases to zero.

Based on our theoretical analysis, we find that, given the parameter γ, there exists an optimal transmission radius $R_\gamma(\lambda)$ that may achieve the maximum information propagation speed $W_\gamma(\lambda)$ in a network with node density λ. However, as we have discussed earlier, actually achieving this maximum speed requires an additional condition that all the relay nodes are aligned and separated from each other by the distance $R_\gamma(\lambda)$. Since the nodes are randomly distributed, it is impossible to find these perfectly located relay nodes when $\lambda < \infty$. There is always a gap between the actually achievable speed $w_\varphi(t)$ and the bound $W_\gamma(\lambda)$.

As the third contribution, we quantify the gap between the actually achievable speed and the desired upper bound. We prove that a packet propagates omnidirectionally in large wireless networks and the speed gap shrinks as the node density increases. Furthermore, we show that in both noise models, there exists a threshold node density, below which the gap is bounded by constants and above which the gap converges to zero exponentially. Since this result is more practical and useful to DTRA missions, we present the main results here.

By definition, the actual information propagation speed is measured by $w_\varphi(t) = \frac{|\mathcal{L}_\varphi(t)|}{t}$. Due to the randomness of node locations, this speed may be faster or slower when the packet travels through different subareas in the network. To evaluate $w_\varphi(t)$ without introducing the subarea bias, we define the *long-term speed* in the direction φ to be

$$w_\varphi = \lim_{t\to\infty} w_\varphi(t) = \lim_{t\to\infty} \frac{|\mathcal{L}_\varphi(t)|}{t}. \tag{48}$$

Since every node uses the same optimal transmission radius $R_\gamma(\lambda)$, the 1-hop transmission time $\tau = \frac{L}{B\log_2(1+\frac{P}{N}R_\gamma^{-\alpha}(\lambda))}$ is the same for every node. Thus, Eq (48) is rewritten as

$$w_\varphi = \lim_{m\to\infty} \frac{Z_m}{m\tau} = \lim_{m\to\infty} \frac{\sum_{i=1}^m \rho_i}{m\tau} = \frac{\overline{\rho}}{\tau}, \tag{49}$$

where $Z_i = |\overline{oz_i}|$, $\rho_i = Z_i - Z_{i-1}$ and $\overline{\rho} = \lim_{m\to\infty} \frac{\sum_{i=1}^m \rho_i}{m} = E[\rho_i]$.

4.1.2 How Fast Can Failure Propagation?

First, we show that the actual information propagation speed is omnidirectional in large networks. In the long term, a packet is disseminated to the same distance away from the source in any direction and the frontier of the covered area is in the circular shape with the source node as the center, as specified in the following theorem.

Theorem 6. *In a network with homogeneous node distributions,* $\forall\ \varphi_1, \varphi_2 \in [0, 2\pi)$, $w_{\varphi_1} = w_{\varphi_2} = w$.

Remark 9. *Theorem 6 states the fact that a packet reaches the same distance away in any direction after sufficiently long propagation time, though it can be faster or slower temporarily in one direction than another.*

Figure 19 depicts an example of the speed comparison in different directions. As the packet propagates farther away, the speeds in all directions converge to the same value.

It can be proved that w is determined by the node density λ, we write $w = w(\lambda) = \frac{\overline{\rho}(\lambda)}{\tau}$. We define the gap between the actual speed $w(\lambda)$ and its upper bound $W_\gamma(\lambda)$ as

$$\varepsilon(\lambda) = \frac{W_\gamma(\lambda) - w(\lambda)}{W_\gamma(\lambda)} = \frac{R_\gamma(\lambda) - \overline{\rho}(\lambda)}{R_\gamma(\lambda)}. \tag{50}$$

The next theorem provides a quantified measurement of $\varepsilon(\lambda)$.

Theorem 7. *In a network where the nodes are randomly distributed in a Poisson point process with density* λ *and the optimal transmission radius* $R_\gamma(\lambda)$ *is used, defining* $a = \lambda\pi R_\gamma^2(\lambda)$, $g_1(a) = \int_0^1 e^{a(x^2-1)}\mathrm{d}x$ *and* $g_2(a) = \int_0^1 e^{-\frac{1}{3}ax^2}\mathrm{d}x$,

$$g_1(a) \le \varepsilon(\lambda) \le g_2(a). \tag{51}$$

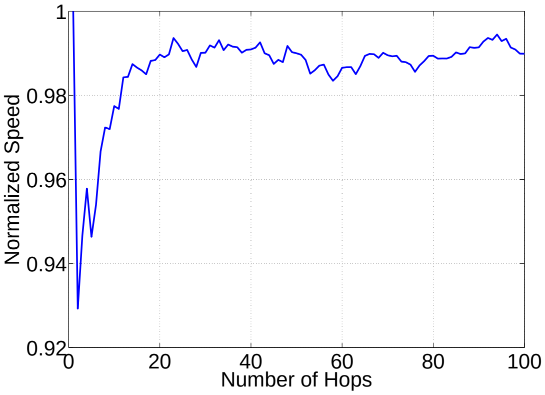
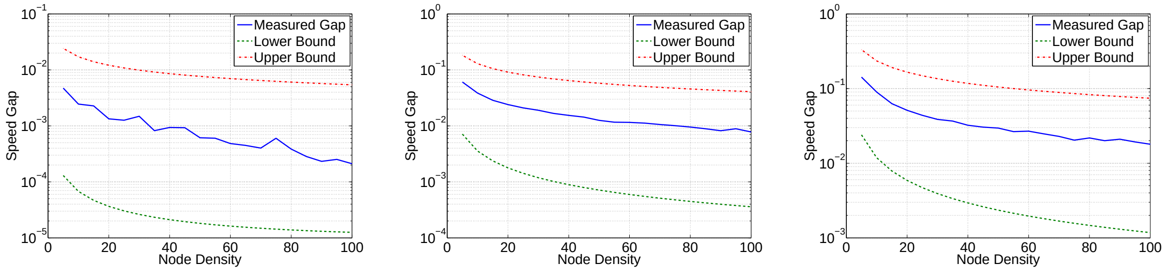


Figure 19: A comparison of the packet propagation speeds in six randomly chosen directions, in which the normalized speed is defined as the ratio of the minimum speed to the maximum speed in the six directions, $\lambda = 30$.



(a) $\alpha = 2$

(b) $\alpha = 4$

(c) $\alpha = 6$

Figure 20: The speed gap in the ambient-noise-dominant model, $\frac{P}{N} = 10^3$.

Remark 10. *Theorem 7 provides the bounds on the convergence rate of the speed gap as the node density increases.*

Remark 11. *We find that in the ambient-noise-dominant model the speed gap converges to zero exponentially with exponent $\lambda^{1-\epsilon}$, where ϵ is an arbitrarily small positive real number. The speed gap $\varepsilon(\lambda)$ and its bounds are shown in Figure 20 as an example.*

Remark 12. *We find that in the interference-dominant noise model the speed gap converges to zero exponentially with exponent $\lambda^{(1-\frac{2}{\alpha})(1-\epsilon)}$, where ϵ is an arbitrarily small positive real number.*

Therefore, our results show that in both noise models there is a threshold node density, below which $\varepsilon(\lambda)$ is bounded by constants (the constants are determined by the choice of parameter γ) and above which $\varepsilon(\lambda)$ converges to zero exponentially, in the rates of $c^{\lambda^{1-\epsilon}}$ and $c^{\lambda^{(1-\frac{2}{\alpha})(1-\epsilon)}}$ respectively.

The study on speed of information propagation is a new problem toward the first thrust of the original goals, that is, the fundamental limits of network vulnerabilities in the present of multiple failures.

4.2 Failure Propagation via Topological Analysis

To assess the health of a network, we may have access to a variety of measurements at the individual nodes. The characterization of these measurements yields a variety of information, which include that about the functionality of the network. To cope with an onset or the aftermath of a significant attack in the likes of one by weapons of mass disruption/destruction, our focus primarily lies in detecting early failures and in localizing them to infer the degree to which the topology of a network is preserved. The power of a detection strategy lies in its versatility and its adaptability

4.2.1 Objectives and Approaches

We take a more abstract approach of characterizing the network processes as 'Data' along with some 'Characteristics' associated with it. We present the task of identifying the process taking place in the network as a Data Classification problem. There is extensive literature present in the field of Statistical Data Learning/Classification which we can put to effective use when modeling the network processes in the above manner. Furthermore, we explore a recently emerging field of Applied Topology to study our network data. Data in many practical applications does not admit any natural geometric properties and searching for metrics in this space might be redundant. (Although, searching for right 'features' which characterize our data is important). It is for this reason that analyzing the topological structure of the data becomes more apt for our situation.

Our approach is to consider a network of nodes represented by X_i and each node has a measurement vector M_i . We assume that the nodes X_i are random samples taken from a manifold and the measurements are samples of a vector field defined on this manifold. There is an operator which is a mapping from this vector field onto a *feature space*. The operator φ is chosen such that the topological invariants of the feature space correspond in some way to the underlying processes on the network. The problem statement then is three-fold:

- Designing the operator based on prior knowledge of the process such that the topology of the feature space reflects the characteristics of the process.
- Identifying or classifying the underlying processes in the network by analyzing the topology, or if necessary and appropriate, the geometry of the feature space.
- Localizing the required processes in the network.

4.2.2 Experimental Results and Observations

We experiment with the network database from the KDD 2000 contest. The data base is a series of records where each record describes a connection. Many attacks/intrusions were emulated in a real-life networks and information about each connection were stored as a record. In our model, we treat the set of records at each node X_i as the measurement vector M_i . Also, as of now, we collect all these measurement vectors and perform "centralized" processing. Our future goal is to decentralize the processing so as to minimize the overhead on the network. In our preliminary attempts towards designing such an operator, we employed the linear discriminant analysis. In this method, we design a linear operator from the measurement space to the feature space such that points corresponding to different processes (“normal” connections, DOS attack, etc...) form different clusters and points within each cluster are packed as tightly as possible. We describe below the linear discriminant analysis briefly:

We are given a training data set X_{Nk} , N is the number of records and k is the dimension of each record. The records correspond to C different processes which we call classes which are available to us. Therefore, the data set can be partitioned into C disjoint sets as $X = X_1, X_2, ..., X_c$ where X_i represents the set of records corresponding to the class i . We represent a record in the set X_i as x^i .The number of records in class i is represented by N_i . The goal is to design a linear operator $\hat{A}$ so as to optimize the *fisher criterion:*

$$\hat{A} = \text{argmax} \frac{A^T S_b \cdot A}{A^T S_w \cdot A} \tag{52}$$

where S_b is called the *between class scatter matrix* and S_w is called the *within class scatter matrix*. The above optimization problem reduces to generalized eigen-value problem and can be solved by simultaneously diagonalizing both the scatter matrices. The figures below show that effectiveness of such analysis.

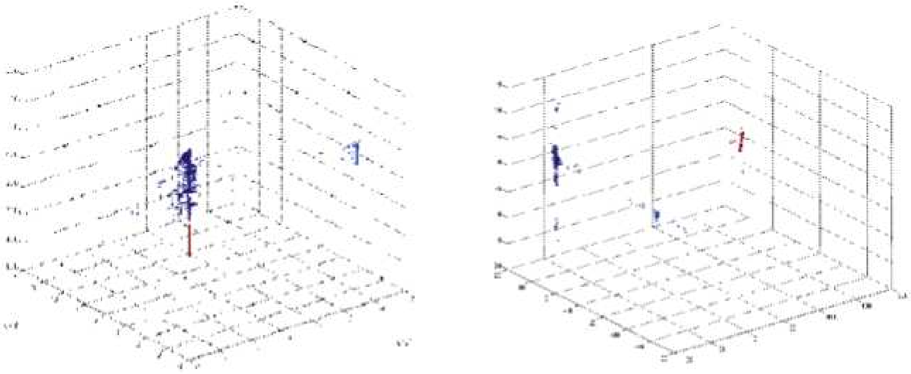


Figure 21: Result of linear discriminant analysis.

To the left above in Figure 21 records from three different processes are plotted in three dimensions (after reducing the dimension from 41 to 3). On the right in Figure21 shows the same records after being mapped onto the feature space using the linear operator designed as stated above. As the figure shows, the records of two kinds of processes which were indistinguishable originally shows considerable amount of topological variation in the feature space. Even though they perform very well in certain cases, the linear operators as design above form a small subset of the vast class of operators one can design and fail to capture the non-linear discriminant features in the measurement data. For this, we should consider the classes of non-linear maps and optimize among those classes.

To give an idea of which non-linear operators are best suited for certain situations, consider the following Figure 22: The figure to the left shows a simulated data set which red ”blobs” show on the top and to the bottom correspond to the same process. The resulting feature space of our mapping should be able to group the red blobs together and separate the blue blob. Clearly, linear operators cannot accomplish such a task. The figure to the right shows the same data points but color coded with the feature values of a non-linear operator. The feature

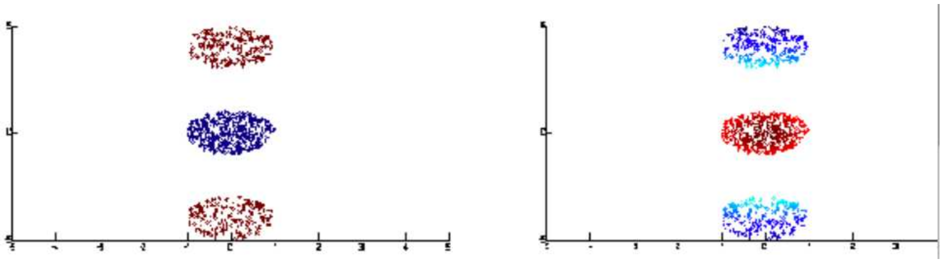


Figure 22: Non-linear mapping.

space is one-dimensional in this case. The mapping of this non-linear operator φ: $\Re \rightarrow \Re$ is shown in Figure 23.

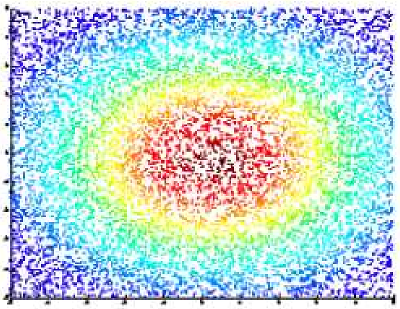


Figure 23: Optimization result with non-linear operator.

It is quite apparent at this stage that non-linear features can better capture the discriminative characteristic of the data. But it is often difficult to choose the right class of non-linear operators. Also, is very important and extremely useful to be able to design such an operator in a decentralized manner within the network without collecting the data at a central place. Both the above problem will be a major focus of our future research.

4.2.3 Clustering Analysis based on Applied Topology

We further studied *Applied Topology*. As seen in the above application, we were looking for different clusters in the feature space to distinguish between different processes. Clustering can be viewed as a discrete version of looking for connected components in a continuous space. Such a feature is called a topological feature. The phrase "studying topology of a space" roughly implies studying the arrangement of points in that space. We extend the above idea of looking for connected components, which is a zero order feature to higher order topological features such as cycles, voids etc. We present here a few applications where such higher order features can give us considerable information about the underlying process on which this topological space is described.

We start with a practical scenario/process and represent it using a *topological space*. Briefly, a topological space describes a set of all open sets so that we can define a notion of continuity. We then analyze these spaces to extract topological invariants which characterize the underlying processes. In the application described above, the underlying processes are the different kinds of network connections and the topological space is the feature space (the range of our designed operator) in which we were searching for connected components. In order to facilitate the digital computation of these invariants, we use *homology* which assigns to each topological space a sequence of algebraic objects such as groups and vector spaces. The topological invariants can be extracted by performing computations on these algebraic objects. The following diagram depicts pictorially the above process.

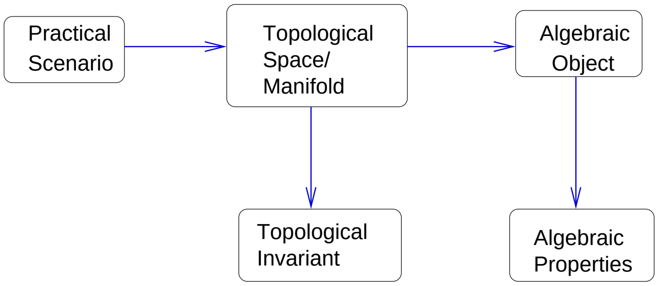


Figure 24: Big picture of applied topology.

Homology is a function which assigns to each topological space, a sequence of algebraic objects. We call these objects homology classes. But computing homology of an uncountable set (containing infinite points such as a disc in two dimensions) is not possible on a computer. We can show that under certain mild conditions we can use random sampling from this space to extract its homology accurately with certain probability guarantees. More specifically, a simplicial complex called the Rips complex formed using these random samples has the same homology classes as the original topological space. We now define precisely a simplicial complex and Homology classes and describe the method of computing these homology classes. We then describe a process for computing these homology classes distributively in a network.

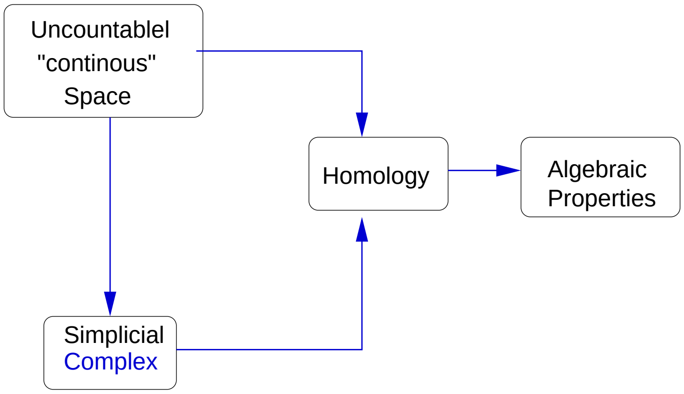


Figure 25: Assigning algebraic objects.

We also investigate other methods such as *simplicial complex* and *chain complex.* A simplicial complex is a generalization of the standard notion of a graph. In a graph we have a set of nodes and a set of pairs of nodes which we call edges. The edges can either be directed or undirected (oriented or unoriented). A Simplicial Complex generalized this concept to form simpleces which can contain any number of nodes instead of two as in edges. A k-th order simplex contains $k+1$ nodes. A face of a k-th order simplex is a $k-1$ order simplex formed by removing one of the node. An oriented k-th order simplex is one which an order is specified for the set of $k + 1$ nodes. A more precise definition of a simplicial complex is as follows:

A simplicial complex K is a collection of simplices σ_i^k where k is the order and i is the index such that:

- Every face of a simplex $\sigma \in K$ also $\in K$.
- The intersection of any two simplices (defined as the simplex with common vertices) is a simplex in K.

For Chain Complex, we assign a sequence of vector spaces C_k to a simplicial complex K . The points in these spaces are called chains. The vector space C_k is formed by taking all the k-th order simplices in K as the basis and taking all linear combinations over a field. We will usually work on the field or real numbers R.

Given a chain complex, we define homology classes (vector spaces in this case) as the following quotient space:

$$H_k(K) = ker(\partial_k)/img(\partial_{k+1})$$

The dimensions of the homology classes give us useful topological information. The dimension of $H_0(K)$ (also called zeroth Betty number or b_0) gives us the number of components. b_1 gives us the number of cycles, b_2 gives us the number of voids and so on.

For computing these classes distributively, we make use of Laplacian operators on the simplicial complex K . The k-th order Laplacian operator $L_k : C_k \to C_k$ is defined as

$$L_k = \partial_{k+1}\partial_{k+1}^* + \partial_k^*\partial_k$$

A special property of the Laplacian operator is that its action over a simplex can be expressed only in terms of its adjacent simplices. We use this local property to compute the laplacian of K over a network. Now, we have an important relation that the kernel of the k-th Laplacian is Isomorphic to $H_k(K)$. Therefore, the problem now reduces to computing distributively the null space of the k-th laplacian.

4.2.4 Relevance to Original Goals

In order to identify failures in a large-scale network, a variety of measurements may be taken. This work, along with some “Characteristics’ associated with it, present a solution for early detection of failures and localization of failures. We present the task of identifying the process taking place in the network as a Data Classification problem in the field of Statistical Data Learning/Classification which we can put to effective use when modeling the network processes. Also, we explored a recently emerging field of Applied Topology to study our network data that do not admit any natural geometric properties.

4.3 Multi-Failure Correlation and Spreading

As one of the two thrusts we intend to explore in this project, we want to understand and characterize the impact of multiple failures caused by WMD attacks on the network infrastructure. In our work of last year, we investigated the multiple failure problem by modeling the network under attack as a devolution process and determined the critical time for network partition. As failures occur over time, the network transits from its original fully connected state into a partitioned state in which node communication distances become limited. Our results give the first evaluation of this critical turning-point in the network devolution process. However, we only considered the scenario of random and independent node failures in last year. The underlying assumption was that multiple node failures are isolated events from one another. We note that in some cases node failures could be correlated in the sense that a failure happens as a result of earlier failures. We extend our previous work by incorporating failure correlations to advance our understanding of the impact of multiple failures on the network infrastructure.

4.3.1 What is Multi-Failure Correlation?

To design WMD resistant networks, we first need to understand the severity of impact that WMD attacks can create in the network infrastructure. Many existing works in the literature have proposed a wealth of solutions that improve the network resilience to failures via different techniques and strategies, including failure prevention, failure detection, failure mitigation, and failure reparation, which aim to restore the normal network functioning when failures happen. The majority of these studies deal with specific types of failures and their countermeasures, for example, planning traffic paths away from the failure-prone regions, providing continuous surveillance coverage when a subset of sensors fail, finding alternative routes when nodes or links become unavailable, and using small-sized packets and error control codes to communicate in networks with low-quality links. These studies have greatly contributed to the failure resilience of large networks. From a different perspective, we intend to characterize the failure impact on the network infrastructure. Our work provides a fundamental understanding on the quantification of failure impact, which is a critical supplement to the existing research works and an important step towards designing effective counter-failure solutions under the threats of WMD.

Our prior work analyzed the critical phase transition time of a large network in which random failures gradually break down an initially connected network into small pieces of disjoint components. The results present an observation on the impact of failures from the network connectivity perspective. However, we did not characterize all the failure possibilities under WMD threats. In some cases, causal relations exist among failures, i.e., some failures happen as a result of other earlier failures caused by WMD. One example of correlated failures is traffic overloading and energy depletion. When a node fails, its traffic is redistributed to the neighboring nodes and these neighboring nodes undertake a heavier packet relaying load than before. Some neighbors may deplete their energy fast and fail in a short time. Another possible case of failure correlation is node malfunction and protocol non-compliance after WMD attacks. When a subset of nodes are damaged by attacks, they may not comply with their designed protocols and cause other nodes to work in unexpected states. The node malfunction may spread out from the initially damaged nodes to other nodes as a result of the protocol non-compliant working states.

In view of the potentially devastating impact caused by correlated failures, we characterize the spread of correlated failures in the network infrastructure. In particular, we attempt to determine the conditions under which a single initial failure will and will not spread to the entire network. In an effort to gain a generally applicable understanding on the spread of correlated failures, we model the failure correlations as general functions and determine their characteristic regimes in terms of the ability of an initial failure to permeate the network. The correlation functions model the geometric constraints in failure propagation, i.e., the distance and probability for a failure to spread in one hop. Based on the correlation functions, we then use the percolation theory to tackle the failure spreading problem. Percolation theory provides a mighty tool for analyzing a wide range of contact-and-relay problems observed in reality. We determine analytically the pervasiveness of failure spreading conditioned on the failure correlation functions by using the concept of percolation. Intuitively, stronger correlations drive an initial failure to spread into a larger area than weaker correlations. Our results provide a quantification of the relation between failure correlations and failure pervasiveness in large networks. The results are useful for us to evaluate and enhance the resilience of our network infrastructure against WMD attacks.

4.3.2 Modeling of Node Failure Correlation

Main Results: Our contributions are the *formal and generalized modeling* of the correlated failures and *determination of the conditions* in which an initial failure will and will not spread out to the entire network.

We model the communication network under WMD threats as a large network consisting of n nodes in a region $\mathcal{B} = [-\frac{L}{2}, \frac{L}{2}]^2$ ($L \rightarrow \infty$). The number of nodes n is assumed to be a Poisson distributed random variable with constant density λ everywhere in the network. Let X_i ($1 \leq i \leq n$) denote the random location of node v_i that is uniformly distributed in the network, independent of n and any X_j ($i \neq j$). We know that $\mathcal{H}_\lambda = \{X_1, \cdots, X_n\}$ is a homogeneous Poisson point process.

We model the node failure correlation by defining two probabilistic correlation functions. When a node fails, it may trigger other failures in its neighbors. Let $\|X_i - X_j\|$ denote the Euclidean distance between v_i and v_j. We define the *failure impact radius* r to be the farthest distance between the location of a triggering failure and the location of an immediate follow-up failure, i.e., failure may propagate from v_i to v_j in one hop only if $\|X_i - X_j\| \leq r_i$, where r_i is the r of v_i. Considering the difference of the nodes in their respective failure impacts, the impact radius r is modeled as a random variable with probability density function $f(x)$ ($0 \leq x \leq 1$). For different nodes v_{i_1} and v_{i_2}, r_{i_1} and r_{i_2} are independent. Besides, we define the *failure connection function* $g(x)$ to model the likelihood of failure propagation from v_i to v_j. If v_j is located within the impact radius of v_i, failure spreads to v_j with a probability $g(\|X_i - X_j\|)$. If v_j is beyond the impact radius of v_i, failure cannot spread from v_i to v_j. For any two pairs of nodes, failure propagation is independent from each other.

To differentiate between the possibilities that a node may be at risk of failure only once and that a node may be impacted by other nodes' failures multiple times, we categorize failure correlations into two classes: *one-time correlation* and *persistent correlation*. We say that the failures are one-time if each node is subject to the impact of other nodes only once. For example, if nodes v_i and v_j have failed sequentially and v_k is within their impact radius r_i and r_j, then v_k only has the failure risk after v_i. If v_k does not fail after v_i, v_k also does not fail when v_j fails. We say that the failures are persistent if a node may fail each time one of its neighbors fails. In the example we used above, v_k may fail after both v_i and v_j with persistent failure correlation.

We use percolation theory to investigate the long-term trend of failure spreading when failures are correlated. When failure propagates from v_i to v_j in one hop, we say that v_i and v_j are connected (or the connection is *open*). Otherwise, v_i and v_j are not connected (or the connection is *closed*). Note that the connections considered here represent the failure correlations among nodes, which are completely different from the communication links. In percolation terminology, each node in the network is a site and failure connections define the bonds

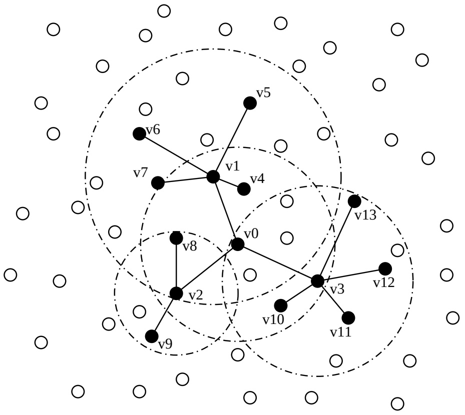


Figure 26: Failure spreading in large wireless networks.

between neighboring sites. As node locations are randomly distributed, this percolation process is also known as *continuum percolation*. In its basic form, continuum percolation assumes an open bond between any two neighboring sites. We have introduced here the probabilistic functions $f(x)$ and $g(x)$ for each bond, so our network model is also called the *random connection model* in which a bond is open only probabilistically.

Given node locations and their bonds, our problem is to determine whether an initial failure will spread to the entire network. An example of failure spreading is illustrated in Figure 26. In this example, the initial failure occurs at node v_0. As a result of this failure, nodes v_1–v_3 fail subsequently and spread the failure further away to nodes v_4–v_{13}. In each step of spreading, a node that has just failed in the previous step passes failure to a random subset of nodes in its neighborhood, as modeled by the impact radius distribution function $f(x)$ and the connection function $g(x)$. As time goes, there are two possible results regarding failure spreading: either the spreading continues for ever or it stops automatically.

We represent the network infrastructure under WMD threats as a *random geometric graph* and denote it as $G(\mathcal{H}_\lambda, f(\cdot), g(\cdot))$. Regarding node v_0 that initiates the failures, we define the *percolation probability* $p_\infty(\lambda, f(\cdot), g(\cdot))$ as

$$p_\infty(\lambda, f(\cdot), g(\cdot)) \triangleq \Pr[|C(v_0)| = \infty], \tag{53}$$

where $C(v_0)$ denotes the cluster of nodes that fail as a result of the initial failure at v_0 and $|C(v_0)|$ denotes the size of cluster $C(v_0)$. When $|C(v_0)| = \infty$, we call $C(v_0)$ a *giant component.* If $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$, failure percolates in the network with a positive probability. If $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$, failure does not percolate in the network almost surely.

Our goal is to determine the regimes of functions $f(x)$ and $g(x)$ such that $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$ and $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$ with given λ, respectively. Obviously, $G(\mathcal{H}_\lambda, f(\cdot), g(\cdot))$ is a subgraph of $G(\mathcal{H}_\lambda, 1, 1)$ and $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$ whenever $p_\infty(\lambda, 1, 1) = 0$. To avoid triviality, we only consider the case $p_\infty(\lambda, 1, 1) > 0$ and determine the respective constraints on $f(x)$ and $g(x)$ for $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$ and $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$. It is well known that there exists a critical density λ_c in $G(\mathcal{H}_\lambda, 1, 1)$ defined as

$$\lambda_c \triangleq \inf\{\lambda > 0 : p_\infty(\lambda, 1, 1) > 0\} \tag{54}$$

that specifies the minimum λ for $p_\infty(\lambda, 1, 1) > 0$. So far, the best known rigorous bounds on λ_c are $0.7698 <$

$\lambda_c < 3.372$. Hence, we assume $\lambda > 3.372$. In addition, if r is a constant, we also assume $\lambda r^2 > 3.372$, which guarantees $p_\infty(\lambda, r, 1) > 0$ by the scaling property.

4.3.3 Impact of One-Time Failure Correlation

Based on our generalized modeling of the failure correlations, we have found the following important results regarding the failure spreading in the network infrastructure after WMD attacks.

We aim to find the conditions under which an initial failure will and will not spread out to the entire network when the failures are one-time correlated. Specifically, we provide the quantified relations among the node density λ, the failure impact radius r and the failure connection probability p that determine the failure spreading trend. In more general cases of the random failure impact radius, we have the following result.

***Theorem* 8.** *With one-time failure correlation, for random failure impact radius r with probability density function $f(x)$ ($0 \leq x \leq 1$) and constant $g(x) = p$,*

i) $p_\infty^{(\text{one})}(\lambda, f(\cdot), p) > 0$ if $ph_1(f(x)) > \frac{\lambda_c}{\lambda}$,

ii) $p_\infty^{(\text{one})}(\lambda, f(\cdot), p) = 0$ if $p(1 - h_2(f(x))) < \frac{\lambda_c}{\lambda}$,

where $h_1(f(x)) = \max_{\{0 \leq a \leq 1\}}\{a^2 \int_a^1 f(x)\mathrm{d}x\}$, $h_2(f(x)) = \max_{\{0 \leq a \leq 1\}}\{(1 - a^2) \int_0^a f(x)\mathrm{d}x\}$.

Intuitively, r and p represent the degree of failure contagiousness. When r and p increase, failure tends to percolate. When they decrease, percolation becomes unlikely to happen. Our results in Theorem 8 present a quantified measure on r and p. Moreover, in the general case of a random r, it indicates that the chance of failure percolation increases if the probability distribution $f(x)$ shifts toward large r (such that $h_1(f(x))$ increases) and decreases if $f(x)$ shifts toward small r (such that $h_2(f(x))$ increases).

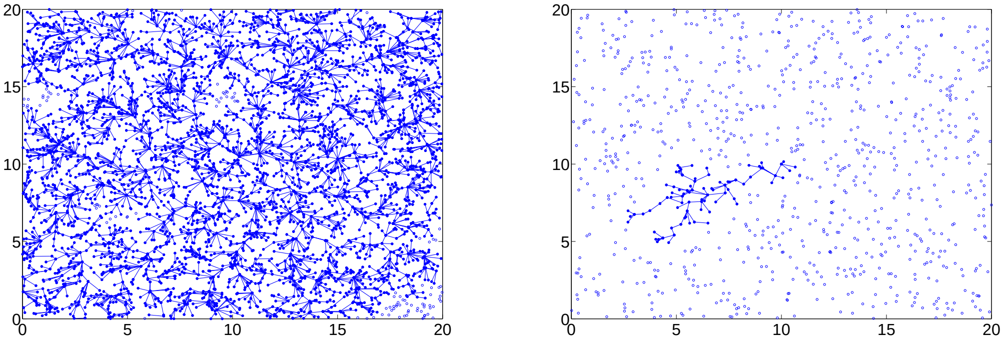
When the failures have persistent correlation, we have derived the following results to characterize the conditions under which failure spreading will and will not happen.

***Theorem* 9.** *With persistent failure correlation, for constant impact radius r and connection probability p, $p_\infty(\lambda, r, p) > 0$ if $pr^2 > \frac{1.8889}{\lambda}$.*

***Theorem* 10.** *With persistent failure correlation, for constant impact radius r and connection probability p, $p_\infty(\lambda, r, p) = 0$ if $pr^2 < \frac{-\ln(1 - \frac{0.1642}{\lambda})}{2.5981\lambda}$.*

Theorems 9 and 10 provide us with the sufficient conditions to judge whether an initial failure will or will not spread to the entire network when the failures are persistently correlated. Our results quantify the relation among λ, r and p to predict the long-term failure spreading trend. We also further generalize the results in Theorems 9 and 10 by considering arbitrary $f(x)$ and $g(x)$ functions and to obtain more applicable results.

As a visual verification of our derived failure percolation conditions, we present the simulation results. Specifically, we give our simulated failure spreading results for Theorem 8. All the other simulation results for the other theorems are similar, so we do not include each of them in this report.



(a) $p\lambda = 9.8$

(b) $p\lambda = 2.3$

Figure 27: Simulation results for failure percolation and non-percolation.

In our simulations, we assume that $f(x)$ is a uniform distribution in the range $0 \leq x \leq 1$. It is not difficult to find $h_1(f(x)) = 0.1481$ and $h_2(f(x)) = 0.3849$. By Theorem 8, we know that failure percolates if $0.1481p\lambda > \lambda_c$ and does not percolate if $0.6151p\lambda < \lambda_c$. Given $0.7698 < \lambda_c < 3.372$, we need to verify the percolation results for $p\lambda > \frac{3.372}{0.1481} = 22.7684$ and $p\lambda < \frac{0.7698}{0.6151} = 1.2515$ respectively.

However, as the existing simulation result has demonstrated that $1.43 < \lambda_c < 1.44$ with high confidence, we simulate $p\lambda > \frac{1.44}{0.1481} = 9.7232$ and $p\lambda < \frac{1.43}{0.6151} = 2.3248$ instead. If failure percolates with $p\lambda > 9.7232$, it must percolate with $p\lambda > 22.7684$. Similarly, if failure percolation does not happen with $p\lambda < 2.3248$, failure definitely cannot percolate with $p\lambda < 1.2515$.

Therefore, we choose $p\lambda = 9.8$ and $p\lambda = 2.3$ respectively with the above consideration. We present the results of failure spread in a 20×20 area in Figure 27, where the initial failure occurs at the center of this area. For clearer presentation, we have only shown in the figures the connections through which failure is propagated from a failed node to a normal node (which becomes failed after the propagation) but ignored the connections between any two nodes that have already failed before these connections are used for failure propagation. We observe that failure is able to spread to a majority of nodes in Figure 27(a) while only to a limited small cluster of nodes in Figure 27(b), hence supporting our claims in Theorem 8.

4.3.4 Impact of Persistent Failures

We explore the percolation of persistent failures in this and next sections. Specifically, we focus on constant failure impact radius r and constant failure connection probability p in this section. We will study the generalized functions $f(x)$ and $g(x)$ in the following section. Next, we discuss separately the sufficient conditions for percolation and non-percolation of persistent failures with constant r and p. We use $p_\infty(\lambda, r, p)$ to denote the percolation probability of persistent failures with constant r and p.

Sufficient Condition of Percolation: Constant r and p Our discussion on the difference between persistent and one-time failures shows that persistent failures are easier to percolate. Therefore, percolation of persistent failures should happen under the same sufficient condition for one-time failures, i.e., $pr^2 > \frac{\lambda_c}{\lambda}$. As the current best known rigorous bounds on λ_c are $0.7698 < \lambda_c < 3.372$, we infer that percolation occurs when $pr^2 > \frac{3.372}{\lambda}$. In regime $pr^2 < \frac{3.372}{\lambda}$, we have no idea on failure percolation. Further judgment depends on the improved accuracy of the bounds on λ_c.

However, since persistent failures are easier to percolate than one-time failures, we expect a tighter sufficient condition that allows percolation with smaller r and p than $pr^2 > \frac{3.372}{\lambda}$. We next determine a new condition by using the technique of continuum-to-discrete percolation mapping. We divide the network area into many small hexagonal cells [53]. As failure spreads, it travels through a cluster of continuous cells. We define a cell as *open* if it contains at least one failed node and *closed* otherwise. Let C_0^{cell} denote the cluster of open cells and $|C_0^{\text{cell}}|$ denote its size. It is obvious that $|C_0^{\text{cell}}| = \infty$ if $|C_0| = \infty$, and vice versa. The mapping between the cluster of failed nodes and the cluster of open cells thus allows us to find the sufficient condition for $|C_0^{\text{cell}}| = \infty$ and use it for $|C_0| = \infty$.

We observe that when the cells have hexagonal shape, the discrete lattice is triangular. In the literature, square cells are usually used as they generate square lattice that is simple and easy for analysis. We use hexagonal cells in this paper for two reasons. First, hexagonal cells yield a tighter bound on r and p for failure percolation than square cells. Second, they render the study of failure non-percolation possible under this mapping framework. With triangular lattice, there exists a critical probability $p_c^{\triangle} = 2\sin(\frac{\pi}{18}) = 0.3473$ [**?**] such that percolation occurs, i.e., $|C_0^{\text{cell}}| = \infty$, if each bond is open with a probability higher than $p_c^{\triangle}$ and not occurs, i.e., $|C_0^{\text{cell}}| < \infty$, otherwise. The discrete percolation in triangular lattice allows us to derive a tighter sufficient condition for failure percolation. Our result is

***Theorem* 11.** *For constant r and p, $p_\infty(\lambda, r, p) > 0$ if $pr^2 > \frac{1.8889}{\lambda}$.*

Sufficient Condition of Non-Percolation: Constant r and p Our result on the sufficient condition of percolation have quantified r and p that are large enough for an initial failure to spread to the entire network. Similarly, we are also interested in determination of r and p when failure cannot percolate in the network. From the continuum-to-discrete percolation mapping, we know that $|C_0| < \infty$ if $|C_0^{\text{cell}}| < \infty$. Thus, our task is to find the condition for $|C_0^{\text{cell}}| < \infty$.

We still divide the network into many hexagonal cells. Compared to square cells, failure non-percolation is easier to study with hexagonal cells. When we choose the cell size sufficiently large, the failed nodes in one cell can only connect to the six neighboring cells and the connections are symmetric in probability. We can focus on the connections between any two neighboring cells to understand failure spreading. With square cells, however, a cell has eight neighbors: four horizontal or vertical ones and four diagonal ones. Its connections to the horizontal or vertical neighbors are not symmetric to those to the diagonal neighbors, rendering failure percolation difficult to analyze. By mapping into hexagonal cells, we have the following theorem to characterize the sufficient condition on r and p for failure non-percolation.

***Theorem* 12.** *For constant r and p, $p_\infty(\lambda, r, p) = 0$ if $pr^2 < \frac{-\ln(1-\frac{0.1642}{\lambda})}{2.5981\lambda}$.*

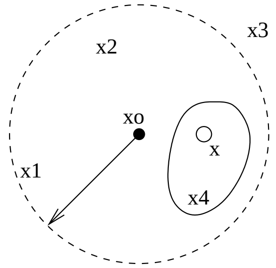


Figure 28: Function $\Psi(g'(x), X_o, \Delta) \triangleq \int_\Delta g'(\|X_o - X_\varepsilon\|)\mathrm{d}\varepsilon$.

Percolation of Persistent Failures: General Failure Correlations When the failure impact radius r is random and the failure connection function $g(x)$ is in a general form, modeling of failure spread becomes even complicated. We present two theorems in this section that characterize the general failure correlations for percolation and non-percolation. To facilitate our study, we first combine the functions $f(x)$ and $g(x)$ into a unified failure connection function.

***Lemma* 4.** *For random r with probability density function $f(x)$ ($0 \leq x \leq 1$) and general $g(x)$, the failure correlations can be modeled equivalently by a constant $r' = 1$ and a unified connection function $g'(x) = g(x)\int_x^1 f(z)\mathrm{d}z$ if $0 \leq x \leq 1$ and $g'(x) = 0$ if $x > 1$.*

We skip the proof of Lemma 4 due to space constraint. Given the equivalence in modeling failure correlations, we consider r' and $g'(x)$ instead of $f(x)$ and $g(x)$ in the rest of this section. For clearer presentation, we now define a few concepts that will be used in our following theorems. We define function

$$\Psi(g'(x), X_o, \Delta) \triangleq \int_\Delta g'(\|X_o - X_\varepsilon\|)\mathrm{d}\varepsilon, \tag{55}$$

which is the integration of the probability $g'(x)$ over region Δ with respect to location X_o, as illustrated in Figure 28. In addition, with respect to all the possible X_o and Δ ($|\Delta| = \sigma$), we define $\Psi_{\min}(g'(x), \sigma) = \min_{\{X_o, |\Delta|=\sigma\}}\{\Psi(g'(x), X_o, \Delta)\}$ and $\Psi_{\max}(g'(x), \sigma) = \max_{\{X_o, |\Delta|=\sigma\}}\{\Psi(g'(x), X_o, \Delta)\}$, which denote the minimum and maximum of $\Psi(g'(x), X_o, \Delta)$ respectively when we change X_o and Δ arbitrarily, as long as the area of Δ is kept constantly as σ. With the help of these definitions, we present our results regarding failure percolation with general $f(x)$ and $g(x)$ as follows.

First, the sufficient condition of percolation regarding $f(x)$ and $g(x)$ is

***Theorem* 13.** *For general $f(x)$ and $g(x)$, $p_\infty(\lambda, f(\cdot), g(\cdot)) > 0$ if $\Psi_{\min}(g'(x), 0.1999) > \frac{0.4266}{\lambda}$.*

Second, the sufficient condition of non-percolation is

***Theorem* 14.** *For general $f(x)$ and $g(x)$, $p_\infty(\lambda, f(\cdot), g(\cdot)) = 0$ if $\Psi_{\max}(g'(x), 1.2535) < \frac{-\ln(1-\frac{0.1642}{\lambda})}{\lambda}$.*

To verify Theorems 13 and 14, we consider a special case of the function $g'(x)$. We let $g'(x)$ be a linearly decreasing function of x, i.e., the probability of failure propagation decreases as the hop distance increases. As before, we generate randomly located nodes in a 20×20 area with node density $\lambda = 5$. The initial failure happens at the center of this area. We first specify $g'(x) = 1 - 0.56x$ ($0 \leq x \leq 1$), which satisfies the condition given in Theorem 13. We also observed that the failure percolates in the network. We then require $g'(x) = 0.0052 - 0.0052x$ ($0 \leq x \leq 1$) which satisfies the condition given in Theorem 14.

Our analysis for failure spread provides a mathematical framework to evaluate the resilience of the network infrastructure against correlated failures. Given a network with node density λ, we may sample the nodes to estimate their failure impact radius distribution $f(x)$ and their failure propagation probability $g(x)$, and determine the characteristic regime (percolation or non-percolation) that λ, $f(x)$ and $g(x)$ fall into. This evaluation allows us to predict the potential impact on the network when failure occurs.

Besides network resilience evaluation, our analysis also indicates a few strategies to prevent correlated failures from wide spreading in the network. Known from the analysis, failure becomes unlikely to percolate if we reduce the impact radius r and the failure connection probability p when the node density λ is given. We are hence able to control the failure spread via bounding r and p.

4.3.5 Relevance to Our Goals

One of our research thrusts is to understand the impact of WMD attacks on the network infrastructure. An insightful understanding of the attack impact provides the foundation for our efforts to design attack-resistant networks. Our work on the failure spreading characterization advances our previous work on the network devolution under random node failures and presents new knowledge regarding the network resilience in an environment with potential WMD threats. Our results have the following unique features:

- We have presented a formal and generalized analytical model to characterize the correlations among a variety of failures. This model captures the essential features of failure correlations.
- For one-time failures, we contain the failure spread by limiting either r or p. To limit r, each node in the network is configured not to execute any suspicious command received from nodes located beyond a certain distance, which effectively reduces the impact radius of each failed node. To limit p, we need to sample and test some portion of nodes in the network to ensure that these tested nodes are not vulnerable to the type of failure that we are concerned with, such that the failure probability of an arbitrary node in the network is controllably low after the test.
- For persistent failures, however, we may not be able to reduce r or p separately. In the example of traffic overloading, if we require the routing logic to limit the path reparation radius to a small value, the failure probability of each node within this radius might be high as each node is expected to receive a large share of the re-routed traffic. In other words, r and p are coupled. Therefore, the best strategy to reduce the degree of failure correlations is to incorporate load balancing into the routing protocol design.
- In addition to the network resilience evaluation, our results also suggest network design criteria to ensure the network architecture to meet certain resilience requirement.

4.4 Inter-Cooperation toward Robust Communications

Since the radio resource in the recovery network is limited, we should make full utilization of the resource to maximize the information delivery capability of the recovery network. Our work on the localized scheduling achieves the maximum utilization goal by coordinating node transmissions carefully to avoid collisions. When each node has a large amount of information to exchange with others, the radio resource in the network is fully used by our scheduling scheme. However, if every node does not have sufficient information to send, the radio resource is wasted when a node is assigned a radio channel but does not send any information over the radio channel. In such cases, we can temporarily allow other nodes to borrow the unused channels to improve the resource utilization, which is called a *Cognitive Radio Network* (CRN). We hence have studied the information propagation problem in cognitive radio networks as a supplementary strategy to maximally exploit the recovery network utilization.

4.4.1 Objectives and Spectrum Recycling

Our research objective is to understand the feasibility and capability of delivering information in the recovery network by using the opportunistically available radio channels. In recent years, there has been intensive research on understanding and optimizing the performance limits in cognitive radio networks. However, an interesting question is still open that what are the *achievable* benefits of information dissemination in such networks, which is essential to the full exploration of the potentials and applications in cognitive radio networks. Understanding how packets disseminate and their temporal and spatial limits, such as dissemination area, transmission speed and latency, can also be beneficial to the deployment, design and application of cognitive radio networks.

Similar problems, on the other hand, have been studied for wireless multihop networks or sensor networks, which explore the conditions for connectivity or percolation in order to ensure the information can be disseminated to the whole network. In addition, information propagation speed or delay has been discussed in recent works, which categorized the delay into bandwidth-incurred propagation delay and topology-incurred delay. In particular, when the network topology remains unchanged or changes very slowly, the bandwidth-incurred propagation delay, which is the transmission time spent by a packet in all the links along its transportation path, is dominant, while topology-incurred delay is negligible.

However, these existing results on multihop wireless networks are not applicable to cognitive radio networks. For instance, the network topology in cognitive radio networks is dynamic not only because of factors such as user mobility and radio link quality, but it is more or less due to the opportunistic channels available over time. As a result, the network is more likely to be a percolated network. That is, a network is almost surely connected, instead of a fully connected one as assumed in earlier works. Furthermore, in current studies on network topology and performance, the critical density is a key condition in identifying whether a network is percolated or not. When node density is higher than the critical density, the network is considered percolated; otherwise, the network is not percolated. The challenge is that prior results are based on a common assumption of homogeneous networks in which all nodes are the same. Nonetheless, a cognitive radio network is intrinsically a heterogeneous network because primary nodes and secondary nodes are different from each other regarding their transmission ranges, usages of communication channels, locations, and routing functions. Therefore, how to determine the conditions for a percolated cognitive radio network is an unknown problem.

Our research approach is to determine the fundamental limits for information propagation in cognitive radio networks by defining new models and new metrics that capture the most basic and important characteristics regarding information propagation. Specifically, we focus on the following two questions: (1) for a large multi-channel cognitive radio network, how far can a packet originated from an arbitrary node be disseminated? (2) When a packet can be disseminated to a sufficiently large area, how long does it take this packet to reach a chosen destination? To tackle these problems, we define two new metrics, the disseminating radius $\|\mathcal{L}(t)\|$ and the propagation speed $\mathcal{S}(d)$ to study the spatial and temporal limits, respectively. The former is the maximum Euclidean distance that a packet disseminates in time t and can be used to characterize the dissemination area. The latter is the speed that a packet transmits between a source and destination at distance d apart, which can be used to interpret the end-to-end delay. Here, we focus on the topology-incurred delay by ignoring bandwidth-incurred propagation delay. Our study has leaded to a few insightful understandings on $\|\mathcal{L}(t)\|$ and $\mathcal{S}(d)$.

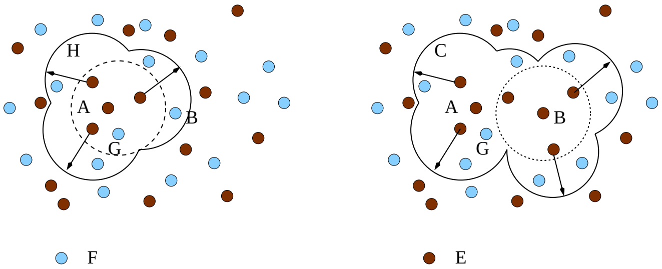
4.4.2 Main Results and Inter-Operation

Main Results: We consider a large cognitive radio network consisting of n secondary nodes $\{v_1, \ldots, v_n\}$, which distribute independently and uniformly in a region $\Omega = [0, \sqrt{\frac{n}{\lambda}}]^2$ for some constant λ and opportunistically access a set of channels $\{ch_1, \ldots, ch_m\}$. Each ch_k is licensed to an overlaid primary network Poisson distributed with density λ_{pk}. Instead of homogeneous transmission range, each v_i is assumed to have an independently adaptive transmission range r_i with $\mathbb{P}(r_i < \gamma)$ for some constant $\gamma > 0$, to save energy and limit interference with primary nodes. We assume that r_i follows a common distribution F_r for simplicity. Let $\mathcal{H}_\lambda = \{X_1, \ldots, X_n\}$ denote the random locations of secondary nodes. Denote $\|X_i - X_j\|$ as the Euclidean distance between v_i and v_j and they may communicate with each other directly only when $\|X_i - X_j\| < \min(r_i, r_j)$. We only consider $\gamma = 1$ and results derived here can be easily extended to the scenarios with any $\gamma > 0$.

In cognitive radio networks, each secondary node v_i needs to frequently sense the communications among primary users in the neighborhood before transmission to avoid interference. That is, each v_i alternates independently between the *communicating* (active) state and *sensing* (inactive) state with periods determined by the stationary *i.i.d* on/off process $W(t)$. Denote η as the stationary probability of v_i being active. Let R_I be the interference range of primary nodes. We say ch_k is *available* to the link v_iv_j if there is no primary users using ch_k within $\mathcal{B}(X_i, R_I) \cup \mathcal{B}(X_j, R_I)$, where $\mathcal{B}(x, r)$ denotes a circle with radius r centering at x. Denote $\mathbb{P}_s$ as the probability that there exist at least one channel *available* to v_iv_j. We represent the cognitive radio network as $G(\mathcal{H}_\lambda, F_r, W(t))$. To address our research problems, we have defined a few concepts.

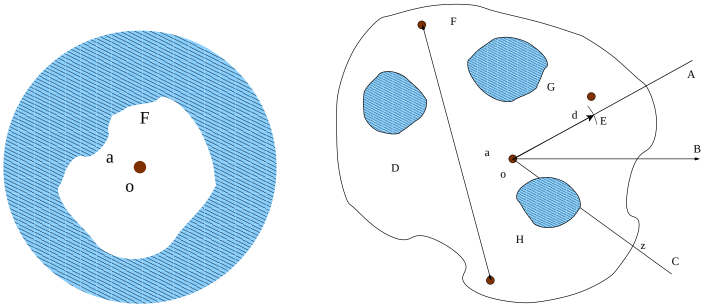
4.4.3 Dissemination Area

We consider that information is disseminated through broadcasting in cognitive radio networks. Let us denote $\mathcal{V}(t)$ as the cluster of nodes that have received the packet by time t, given that the packet is sent at time $t = 0$. The *dissemination area* at t, $\mathcal{A}(t) \in \mathbb{R}^2$, that is, the total area covered by $\mathcal{V}(t)$, can be expressed by $\mathcal{A}(t) \triangleq \bigcup_{v_i \in \mathcal{V}(t)} \mathcal{B}(X_i, 1)$, where $\mathcal{B}(x, r)$ is a ball with radius r centering at point $x \in \mathbb{R}^2$ and X_i is the location of v_i.



(a) t = 0 (b) t= τ

Figure 29: Illustration of information dissemination in a percolated CR network.



(a) $\mathcal{A}_\infty < \infty$ (b) $\mathcal{A}_\infty = \infty$

Figure 30: An illustration of dissemination area after long time t.

An illustration of information dissemination in a cognitive radio network $G(\mathcal{H}_\lambda, F_r, W(t))$ is shown in Figure 29 in which a packet is originated by node v_0. When v_0 starts broadcasting this packet at time 0, its neighbors receive the packet and they rebroadcast the packet. Here two nodes v_i and v_j are called "neighbors" at $t = t'$ if the link $v_i v_j \in G(\mathcal{H}_\lambda, F_r, W(t'))$. Without considering propagation delay, at $t = 0$, the packet has spread to cluster $\mathcal{V}(0) \subset G(\mathcal{H}_\lambda, F_r, W(0))$ containing v_0 (see Figure 29(a)). We refer τ as *topology-incurred delay* for node $v \in \mathcal{V}(\tau) \backslash \mathcal{V}(\tau^-)$ to receive this packet from v_0.

Assume that v_0 is located at the origin $o \in \mathbb{R}^2$. Illustrations of dissemination area for sufficiently large t are shown in Figure 30. Since the network is not fully connected, any sufficiently large area $\mathcal{B}$ is only partially covered by $\mathcal{A}(t)$. The uncovered area, which is also called *vacancy* in this paper, is shown in Figure 30 as shaded areas. Letting $\mathcal{A}_\infty = \lim_{t \to \infty} \mathcal{A}(t)$, we illustrate $\mathcal{A}_\infty < \infty$ and $\mathcal{A}_\infty = \infty$ in Figure 30(a) and Figure 30(b) respectively. Denote $\mathcal{L}_\varphi$ as the line starting from the origin o in the direction $\varphi \in [0, 2\pi)$ and $\mathcal{L}_\varphi(t) = oz$, where $z = \arg\max_{v \in \mathcal{L}_\varphi \cap \mathcal{A}(t)} \|v\|$ is the farthest intersection point between $\mathcal{L}_\varphi$ and $\mathcal{A}(t)$. For example, in Figure 30(b), $\mathcal{L}_{\varphi 1}(t)$ is the line segment $\overline{oz_1}$. The length of $\mathcal{L}_\varphi(t)$, $\|\mathcal{L}_\varphi(t)\|$, is defined as the *transmitting distance* at t.

Definition 7 (Dissemination radius $\|\mathcal{L}(t)\|$)**.** *The* dissemination radius *at time t is defined as* $\|\mathcal{L}(t)\| = \max_{\varphi \in [0, 2\pi)} \|\mathcal{L}_\varphi(t)\|$. *The* limiting dissemination radius *is defined as* $\|\mathcal{L}(\infty)\| = \lim_{t \to \infty} \|\mathcal{L}(t)\|$.

The *dissemination radius* indicates how *far* a packet can reach in spatial domain in a large network. Next, we move on to the temporal domain. Given $\mathcal{L}_\varphi(t)$, an intuitive definition of information propagation speed is $\frac{\|\mathcal{L}_\varphi(t)\|}{t}$, which tells the maximum speed in direction φ. However, due to the dynamics in cognitive radio networks, a node closer to v_0 may receive the packet later than a farther node. For example, in Figure 29, v_2 receives the packet later than v_1. Instead of maximum speed, we are more interested in how long the packet can be disseminated to a chosen destination at d apart. Thus, the *Information Propagation Speed* is defined as follows.

Definition 8 (Information propagation speed $\mathcal{S}_\varphi(d)$)**.** *Let $\mathfrak{r}_\varphi(d)$ be the point on $\mathcal{L}_\varphi$ with $\|\mathfrak{r}_\varphi(d)\| = d$ (see Figure 30). Denote $\mathcal{T}(v_0, v) \triangleq \arg\min_{t\geq 0}\{v \in \mathcal{V}(t)\}$ as the topology-incurred delay of the node v. Denote $\tilde{v}_\varphi(d)$ as the node closest to $\mathfrak{r}_\varphi(d)$ which can receive the packet. That is, $\tilde{v}_\varphi(d) = \arg\min_{v\in\mathcal{V}_\infty} \|v - \mathfrak{r}_\varphi(d)\|$, where $\mathcal{V}_\infty = \lim_{t\to\infty} \mathcal{V}(t)$. When $\|\mathcal{L}(\infty)\| = \infty$, the* Information Propagation Speed *in direction φ is defined as $\mathcal{S}_\varphi(d) \triangleq \frac{d}{\mathcal{T}(v_0,\tilde{v}_\varphi(d))}$. When $\|\mathcal{L}(\infty)\| < \infty$, $\mathcal{S}_\varphi(d) \triangleq \frac{d}{\mathcal{T}(v_0,\tilde{v}_\varphi(d))}$ for $d \leq \|\mathcal{L}(\infty)\|$, and $\mathcal{S}_\varphi(d) \triangleq 0$ for $d > \|\mathcal{L}(\infty)\|$. The* limiting propagation speed *$\mathcal{S}_\varphi(\infty)$ is defined as $\lim_{d\to\infty} \mathcal{S}_\varphi(d)$. The definition of $\mathcal{S}(d)$ denotes the propagation speed in an arbitrary direction.*

For convenience of presentation, we have the following terminologies, which differentiate *infinite* and *finite* limits of information dissemination.

Definition 9. *For a packet b originated by v_0 at $t = 0$, b is said to be disseminated* locally *if $\|\mathcal{L}(\infty)\| < \infty$ and* globally *otherwise. Particularly, b is said to be disseminated* globally "instantaneously" *if $\|\mathcal{L}(0)\| = \infty$, be disseminated* globally "within finite time" *if $\|\mathcal{L}(0)\| < \infty$ but $\|\mathcal{L}(\phi)\| = \infty$ for some bounded ϕ, and be disseminated* "gradually" *if $\mathcal{L}(t) < \infty$ for any t but $\mathcal{L}(\infty) = \infty$.*

Based on our models and definitions, we have discovered the following important results regarding the information propagation radius and speed.

***Theorem* 15.** *For a cognitive radio network $G(\mathcal{H}_\lambda, F_r, W(t))$, given the number of channels m and the spatial density of primary nodes $\lambda_p = \{\lambda_{pk}\}_{k=1}^m$, there exists a critical density $\lambda_{c,c}$ on secondary nodes, above which $G(\mathcal{H}_\lambda, F_r, W(t))$ remains percolated for all t and below which $G(\mathcal{H}_\lambda, F_r, W(t))$ is not percolated for all t. Specifically, we have*

$$\lambda_{c,c} < \min_{0<\|e\|<0.5} \frac{1.21\log\left(\frac{1}{1-\sqrt{\frac{\mathbb{P}_c}{\mathbb{P}_s}}}\right)}{\|e\|^2\eta(1-F_r(2\|e\|))}, \tag{56}$$

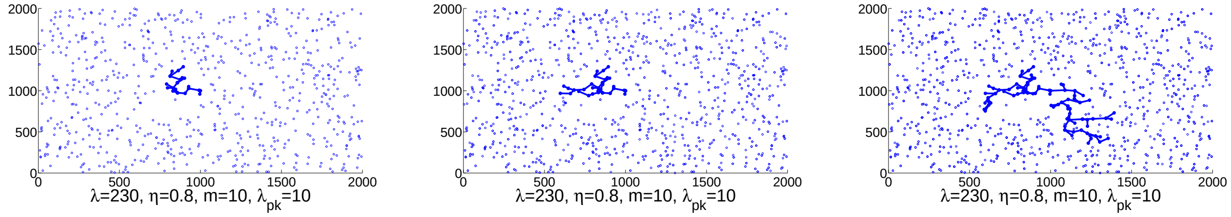
where $\mathbb{P}_c$ is the bond open probability sufficient for percolation on a dependent triangular lattice and $\mathbb{P}_s = 1 - \prod_{k=1}^m \left(1 - e^{-\lambda_{pk}\alpha}\right)$. Furthermore, almost surely,

$$\lambda_{c,c} > \frac{1}{\Gamma(1-\mathcal{C}_{LCC})}, \tag{57}$$

where $\Gamma = 2\pi\eta\int_0^1\int_0^r x(1-F_r(x))\mathbb{P}_s dx dF_r$ and $\mathcal{C}_{LCC}$ is the Link Correlation Coefficient.

***Theorem* 16.** *(i) When $\lambda_{c,w} < \lambda < \lambda_{c,c}$ and $v_0 \in \mathbb{C}_\infty(G(\mathcal{H}_\lambda, 1))$, information b originated by v_0 can be disseminated* gradually *at some constant speed $\mathcal{S}_\varphi(d) = \kappa$, for sufficiently large d. (ii) When $\lambda > \lambda_{c,c}$ and $v_0 \in \mathbb{C}_\infty(G(\mathcal{H}_\lambda, 1))\backslash\mathbb{C}_\infty(G(\mathcal{H}_\lambda, F_r, W(0)))$, b can be disseminated* globally within finite time *with probability 1. (iii) When $\lambda < \lambda_{c,w}$, b can only be disseminated* locally *with probability 1 and the limiting speed $\mathcal{S}_\varphi(\infty) = 0$.*

Figure 31 shows an example of when $\lambda_{c,w} < \lambda < \lambda_{c,w}$, how the packet b originated from v_0 disseminates, where the bigger dots connected by the solid line denote the cluster of nodes that have received b at time t.

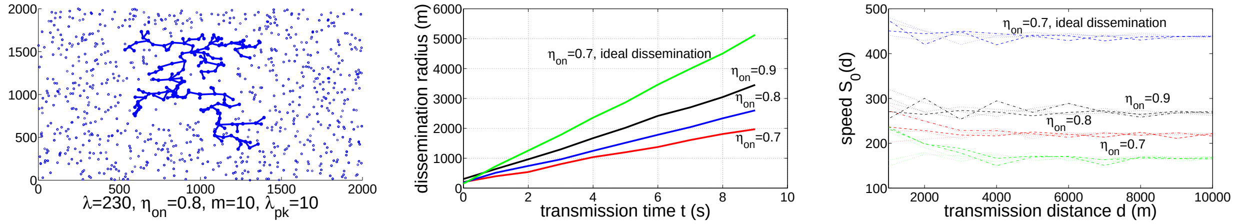


(a) t = 0s

(b) t = 1s

(c) t = 2s

Figure 31: Snapshots of packets disseminating gradually when $\lambda_{c,w} < \lambda < \lambda_{c,c}$.



(a) t = 2s, ideal dissemination

(b) Dissemination radius $\mathcal{L}(t)$

(c) Propagation speed $\mathcal{S}_{\varphi}(d)$

Figure 32: Dissemination radius and propagation speed analysis.

Particularly, given $\mathcal{T}_{on} = 0.9s$, $\mathcal{T}_{off} = 0.1s$ and $\lambda = 230$ (per km^2), we find that $G(\mathcal{H}_\lambda, 1)$ is percolated but $G(\mathcal{H}_\lambda, F_r, W(t))$ not by simulation. As expected, in this case, only a very small set of nodes will receive b initially, as shown in Figure 31(a). However, b will be disseminated to more and more nodes *gradually*, as shown in Figure 31(b) and Figure 31(c).

For comparison, we also consider an ideal dissemination strategy, which has been well studied in the wireless sensor networks, by assuming that once a secondary node receives b, it stays *active* and keeps sending b using maximum transmission power, until all nodes within its maximum transmission range receive b. Comparing with Figure 31(c), Figure 32(a) shows that, with ideal dissemination strategy, the information disseminates at least two times faster.

The average dissemination radius based on 100 independent simulations with parameters $\lambda = 230$ (per km^2), $m = 10$ and $\lambda_{pk} = 10$ (per km^2) is shown in Figure 32(b). We find that the dissemination radius using ideal dissemination strategy almost scales linearly with time. An interesting observation is that the dissemination radius without using ideal dissemination strategy still increases linearly with t for any η, although the speed is much slower. We also note that the dissemination radius is an *increasing* function of η. This is natural since larger η means more nodes in *communicating* state, which can help relay the packet farther. 5 independent simulations of propagation speed $\mathcal{S}(d)$ with parameters $\lambda = 230$ (per km^2), $m = 10$ and $\lambda_{pk} = 10$ (per km^2) are shown in Figure 32(c). We observe that although the practical information dissemination in cognitive radio networks is much slower than that in wireless sensor networks with ideal dissemination strategy, the propagation speed $\mathcal{S}(d)$ is still some constant for large transmission distance d. This validates our theoretical results in Theorem 16. Similar to Figure 32(b), we also observe that $\mathcal{S}(d)$ can be increased by increasing η.

4.4.4 Importance of Inter-Operation Networks

One of our original research objectives is to design recovery strategies for fast communication restoration after WMD attacks. As the recovery network is limited in available radio resources, we should maximally exploit the radio utilization in the recovery network in order to deliver the large amount of information that enters the network after an attack. Our work on the cognitive radio networks provides the fundamental understanding on the possibility of utilizing temporarily unused radio channels for additional information delivery, which is an important and promising technique to achieve the full network utilization in the recovery network. To be specific, our work has contributed to our original research goals in the following few perspectives.

- We have constructed a network model that captures the important communication features in cognitive radio networks in respect to the information propagation process.
- We have defined the novel metrics to investigate quantitatively the radius and speed of information propagation in the cognitive radio networks.
- We have derived the conditions governing information propagation capability and characterized the information propagation radius and speed in cognitive radio networks.

4.5 Failure Propagation in Inhomogeneous Networks

Our previous study along with other literature on performance limits in cognitive radio networks demonstrates the feasibility of information delivery through recycling the temporarily unused radio resources in the recovery network, like in cognitive radio networks (CRNs)k. Since the recovery network is limited in the availability of radio resources, the cognitive radio technology is important to supplement the scheduling schemes to make the full network utilization. Therefore, an interesting, yet challenging question is that *how fast can failures propagate* in such an inhomogeneous networks. There are three main challenges in addressing this question. First we must consider that there are two types of users, the primary users and secondary users, in which primary users who have license from the regulator and thus have priority to utilize spectrum, and secondary users who opportunistically access spectrum without interfering with the coexisting primary users. This demands for a new study in an *inhomogeneous network* instead of *homogeneous networks* in the past. Second, we consider finite as well as large CRNs, where secondary users are mobile under general mobility and primary users are either mobile or static. In other words, we need to provide a general mobility framework which captures most characteristics of the existing mobility models and takes spatial heterogeneity into account. Third, we must consider both “finite” and “large” networks,” while most of existing works focus on asymptotic results for *large* or infinite large networks.

4.5.1 Objectives and Approaches

By reviewing existing studies on CRN performance, we find that the seminary in[52, 54] studied the packet latency in the *fully connected* wireless ad-hoc networks and showed that there exist bounds on the latency and these bounds are tight when the number of nodes are large enough. Instead of *full connectivity*, the work in [55, 56] further showed that the latency scales asymptotically at least linearly with the transmission distance in wireless sensor networks when these networks are *percolated.* Although these results have greatly advanced our understanding of the nature of latency, they may not be applicable to CRNs for the following challenges.

First, these results were obtained by assuming that wireless nodes are static. However, in many cognitive applications (e.g., cognitive vehicular networks, military networks), secondary users usually need to move around to achieve "better spectrum opportunities" or "more security". In recent years, a significant effort has been devoted by the research community to understand how the mobility influences the performance of wireless ad hoc networks or sensor networks. In the seminal work [57], Grossglauser and Tse showed that mobility can improve the capacity in large wireless ad hoc networks at the cost of the delay. The mobility pattern considered by them assumes that nodes move according to an ergodic process that is equally likely to visit any portion of the network area. Motivated by [57], capacity-delay trade-offs have been extensively studied under various mobility models, such as under the i.i.d model [58], the Browian motion [59], the reshuffling model [60] and different variants of random walks and random way-point [61, 62]. In all these studies, nodes are assumed to be *spatially homogeneous.* That is, the motion of a node uniformly covers the entire network. Later on, spatial inhomogeneity has been taken into account. In [63], the network is partitioned into cells and nodes are restricted to move within on randomly chosen cell. The work in [64] studied the capacity under mobility where each node has a *home point* and the nodes move around their own home points. These works demonstrated that mobility plays an important role in networks' performance, but the question about *how the general mobility (instead of a specific mobility model) affects the performance, or especially the latency of wireless networks* remains open.

Furthermore, the works in [65, 54, 55, 56] only derive the latency of networks where the number of nodes is infinite or approaches to infinity. Although these results enable the deployment of the large general purpose ad hoc or sensor networks, in many real applications, the number of nodes is small and finite. The question of *What the latency is in networks with finite nodes* leaves unanswered. Moreover, the network topologies in [65, 54, 55, 56] are homogeneous, i.e., nodes of these networks are identical. Nevertheless, there exist two types of nodes in CRNs and cognitive (secondary) communications are subject to primary communications. *What the impact of the primary communications on the latency of secondary users is* stay under-explored. We remark that we are not the first to study the heterogeneous topology of CRNs. For example, in [66], Ren *et al.* studied the challenge due to the heterogeneous nodes on connectivity of CRNs. However, to the best of my knowledge, there is little work on the impact of topological heterogeneity on the latency of CRNs.

Therefore, we will study the latency in *general* mobile CRNs. Particularly, we first define an abstract framework which captures most of the features of the existing mobility models and takes *spatial inhomogeneities* into account. Then we study the latency of a CRN where secondary users are mobile under this general framework (and primary users may be either mobile or static). We show that in finite CRNs, the dissemination latency depends on the spatial distribution and the *mobility capability* α (characterizing the region that a mobile user can reach) of secondary users. And given any spatial distribution of secondary users, there exists a *critical* value on α, above which the latency has a heavy tail; and below which the tail of its distribution is bounded by some *Gamma* distributions. In addition, as the network grows to infinity, the latency asymptotically scales linearly with respect to the "distance" (characterized by the transmission hops or Euclidean distance) between the source and destination nodes if the network remains "connected" (fully connected or percolated). Moreover, we further find that although the primary traffic will *negatively* impact the expected latency, it will not influence the "threshold" structure (on α)of the distribution of the latency in finite CRNs and the "linearity" of the asymptotic latency (with respect to the dissemination "distance") in large "connected" CRNs.

4.5.2 Main Results of Spatial and Temporal Limits

Our contributions on this works are three-fold: a general mobility model, latency for finite inhomogeneous networks, and large, percolated networks.

Mobility models: We consider a family of mobility models which are denoted by $\mathcal{M}(\Phi, \Psi, \alpha)$ and characterized by three parameters Φ, Ψ, α. *Spatial heterogeneity* has been taken into account in $\mathcal{M}(\Phi, \Psi, \alpha)$. Specifically, we first study *spatial inhomogeneity* of the trajectory of a particular mobile node. We consider the scenario that a node spends most of the time in a small region, and rarely visits the areas far away from it. We model this behavior by assuming that each node v_i has a *home point* [64], located at v_i^h. Nodes move "around" their home points according to independent stationary and ergodic processes. Moreover, we describe the probability density of a node v_i around v_i^h by a non-increasing and direction-invariant function $\Psi_i(x) = \Psi(x - v_i^h)$. We assume that Ψ_i is non-zero in and only in a region characterized by a constant α; that is, $\Psi_i(x) = \Psi(x - v_i^h) > 0$ when $\|x - v_i^h\| < \alpha$ and $\Psi_i(x) = \Psi(x - v_i^h) = 0$ otherwise. α is called ***mobility capability*** since 2α characterizes the *moving diameter* of nodes.

To further account for *spatial inhomogeneity* of the *home points* over Ω_n, we assume that each home point v_i^h is associated with a fixed point v_i^c, which is called the *center point* of v_i. The center points are regularly placed in Ω_n. For example, $\{v_1^c, \dots, v_n^c\}$ are placed regularly at positions $(\frac{1}{2\sqrt{\lambda}} + \frac{i}{\sqrt{\lambda}}, \frac{1}{2\sqrt{\lambda}} + \frac{j}{\sqrt{\lambda}})$ with $0 \leq i \leq \sqrt{n} - 1$ and $0 \leq j \leq \sqrt{n} - 1$ (see Figure 33). We describe the distribution of the home point v_i^h around v_i^c by a non-increasing probability density function $\Phi_i(x) = \Phi(x - v_i^c)$, which is assumed to be invariant in all directions.

Discussion: "Home points" have been introduced in [64] to describe the *spatial inhomogeneity* incurred by the mobility of a particular wireless node. In $\mathcal{M}(\Phi, \Psi, \alpha)$, besides the "home points", we further introduce the "center points" to model the heterogeneously spatial distribution of the home points, which characterizes the *spatial inhomogeneity* incurred by *heterogeneous mobility* of variant users. This two-level mobility model accounts for a wide range of mobility patterns. For example, if the probability density function $\Phi(x)$ is a constant function independent of x (i.e., home points are uniformly distributed over Ω_n), $\mathcal{M}(\Phi, \Psi, \alpha)$ reduces to *Uniform Anisotropic* model in [64]. Furthermore, if the probability density function $\Psi_i(x) = \Psi(x - v_i^h) = \delta(x - v_i^h)$, where $\delta(x)$ is the Dirac impulse function, $\mathcal{M}(\Phi, \Psi, \alpha)$ reduces to the static model in [67], where nodes are assumed to be static and uniformly distributed; if $\Psi(x)$ is also a constant function independent of x and α, $\mathcal{M}(\Phi, \Psi, \alpha)$ reduces to the homogeneous mobility model in [57]; and if $\Psi(x)$ is a threshold function whose value is zero when $x > \alpha$ and a nonzero constant when $x < \alpha$, $\mathcal{M}(\Phi, \Psi, \alpha)$ reduces to the *constrained i.i.d* model used in [56].

In this work, we assume that **secondary users are mobile under the general mobility pattern** $\mathcal{M}(\Phi, \Psi, \alpha)$. To facilitate the study of the dissemination latency of secondary users, we further categorize $\mathcal{M}(\Phi, \Psi, \alpha)$ into three classes based on the extent of *spatial inhomogeneity* of home points:

- Extremely Inhomogeneous Home Points (**EIHP**) mobility $\mathcal{M}(\Phi_E, \Psi, \alpha)$: Home points are fixed and reg-

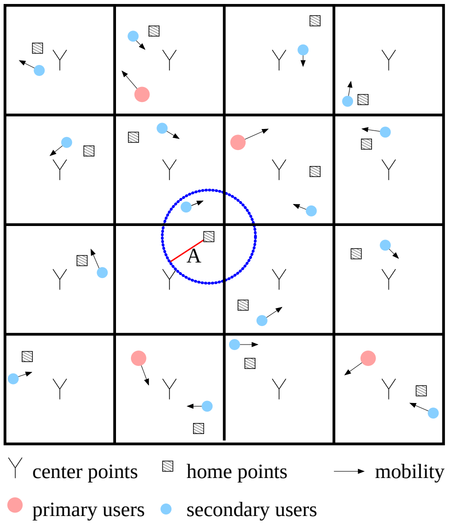


Figure 33: An illustration of the general mobility pattern $\mathcal{M}(\Phi, \Psi, \alpha)$.

ularly placed over Ω_n. Here $\Phi_E(x) = \delta(x)$.

- Partial Inhomogeneous Home Points (**PIHP**) mobility $\mathcal{M}(\Phi_P, \Psi, \alpha)$: As shown in Figure 33, center points $\{v_i^c\}_{i=1}^n$ partitioned Ω_n into n *subregions* $\{\mathcal{O}_i\}_{i=1}^n$ as Voronoi diagrams (we generally assume that n is a square of some integer for simplicity). In this category, the home point v_i^h is independently distributed in $\mathcal{O}_i$. The *clustered grid mobility* in [64] falls into this category.

- Homogeneous Home Points (**HHP**) mobility $\mathcal{M}(\Phi_H, \Psi, \alpha)$: Home points $\{v_i^h\}_{i=1}^n$ are independently and uniformly distributed over Ω_n. Here $\Phi_H(x)$ is a constant density function independent of x.

We need to remark here that between the extremely inhomogeneous mobility **EIHP** and homogeneous mobility **HHP**, there still exist some other partially inhomogeneous mobility patterns, besides **PIHP** considered in this paper. However, the analysis techniques and results for **PIHP** can be easily extended to other partially inhomogeneous mobility patterns. Thus in this study, we only consider **PIHP** case.

Notation and Problem Formulation We first denote $(\mathcal{F}_{m,n}, \mathcal{M}(\Phi, \Psi, \alpha), \mathcal{M}_H)$ as a mobile CRN $\mathcal{F}_{m,n}$ where secondary users are mobile under $\mathcal{M}(\Phi, \Psi, \alpha)$ and primary users are mobile under $\mathcal{M}_H$ throughout the paper. Denote $\mathbb{L}(t)$ as the set of communication links among secondary users in $(\mathcal{F}_{m,n}, \mathcal{M}(\Phi, \Psi, \alpha), \mathcal{M}_H)$ at time t. The interference model [68] shows that $\mathbb{L}(t)$ are dynamic as the primary and secondary users are mobile.

As our main interest lies in the *dissemination latency*, i.e., how fast information can be disseminated from the source secondary user to the destination secondary user, rebroadcasting and "store-carry-and-forward" communication paradigm (also named mobility-assisted routing) have been considered. Specifically, without considering the *propagation delay*, when the source v_s broadcasts a message at time 0, all the secondary users connected to v_s (i.e., secondary users which have a communication path to v_s in $\mathbb{L}(0)$) receive the message instantaneously. The destination v_d may not receive this message if it is disconnected from v_s at time 0. As time goes on, nodes move, and the message is passed from message-carrying secondary users to other secondary users whenever they are connected in $\mathbb{L}(t)$. Thus, the message is disseminated throughout the network and v_d may receive this message at some time as this process continues. Before the problem formulation, we define a few relevant concepts.

Definition 10. *Let $l_{i,j}$ denote a communication link between secondary users v_i and v_j. The* first hitting time *between v_i and v_j is defined as $\mathcal{T}_h(v_i, v_j) \triangleq \inf\{t \geq 0 : l_{i,j} \in \mathbb{L}(t)\}$.*

Definition 11. *Let $\mathcal{V}(t)$ be the set of secondary users that have received the message at time t. The* dissemination latency *between v_s and v_d is defined as $\mathcal{T}_d \triangleq \inf\{t \geq 0 : v_d \in \mathcal{V}(t)\}$.*

$\mathcal{T}_d$ can be coupled as *the first passage time* in the weighted graph [55]. Next we formulate the latency problem addressed in this paper as follows.

Definition 12. Dissemination Latency $\mathcal{T}_d$. *Given $(\mathcal{F}_{m,n}, \mathcal{M}(\Phi, \Psi, \alpha), \mathcal{M}_H)$ with a source secondary user v_s disseminating a message to the destination v_d at time 0, find out:*

1. *In a finite $\mathcal{F}_{m,n}$, what the distribution of the dissemination latency $\mathcal{T}_d$ is;*

2. *as the number of users m and n increases to infinity, the dissemination latency $\mathcal{T}_d$ is scalable or not.*

To further describe "how fast" information can be disseminated, we usually scale the dissemination latency $\mathcal{T}_d$ with the "distance" between the source and destination secondary users.

Definition 13. Distance $\mathcal{D}$. *In $(\mathcal{F}_{m,n}, \mathcal{M}(\Phi, \Psi, \alpha), \mathcal{M}_H)$, three metrics can be used to characterize how far two nodes v_i and v_j are apart: the "distance" between secondary users v_i and v_j at time t $d^{(t)}(v_i, v_j)$, the "distance" between v_i^h and v_j^h $d_h(v_i, v_j)$, and the "distance" between v_i^c and v_j^c $d_c(v_i, v_j)$. Here the "distance" can be any* p-norm *metric function and we consider two of the most popular metrics: "transmission hops" and Euclidean distance. Denote $\mathcal{D}$ as the "distance" between v_s and v_d and define $\mathcal{S}_d \triangleq \frac{\mathcal{T}_d}{\mathcal{D}}$. $\mathcal{S}_d$ characterizes how fast information disseminates and is called "dissemination speed" in this paper for convenience. $\mathcal{D}$ will be specified in the particular analysis.*

The key question in this study is how fast information is disseminated in both finite and large CRNs under general mobility $\mathcal{M}(\Phi, \Psi, \alpha)$. To facilitate our analysis, we first study the dissemination latency $\mathcal{T}_d$ in CRNs where secondary users are mobile under the three subclasses of models **EIHP**, **PIHP** and **HHP**, respectively. Then based on the generalization of these results, we obtain the fundamental properties of the dissemination latency $\mathcal{T}_d$ when secondary users are mobile under the general mobility $\mathcal{M}(\Phi, \Psi, \alpha)$. We summarize our main results as follows.

***Theorem* 17.** *In a finite CRN $(\mathcal{F}_{m,n}, \mathcal{M}(\Phi, \Psi, \alpha), \mathcal{M}_H)$, there exists a* **critical value** *on the* mobility capability *α, above which the tail of dissemination latency $\mathcal{T}_d$ is bounded by some* Gamma *distribution; below which $\mathcal{T}_d$ has a* heavy-tailed *distribution and $\mathbb{P}(\mathcal{T}_d = \infty) > 0$.*

Remark 13. *$\mathbb{P}(\mathcal{T}_d = \infty) > 0$ indicates a positive probability that the destination will not receive the message from the source. Thus the requirement $\mathbb{P}(\mathcal{T}_d < \infty) = 1$ in the mobile wireless networks is equivalent to the* connectivity *in the wired networks, which is used as a prerequisite to evaluate the functionality of the network applications. Moreover, a* heavy tail *of the dissemination latency $\mathcal{T}_d$ implies a significant probability that it takes long time to disseminate a message from the source to the destination. Thus not only a bounded dissemination latency (i.e., $\mathbb{P}(\mathcal{T}_d < \infty) = 1$), a* light-tailed *distributed dissemination latency $\mathcal{T}_d$ (i.e., $E(\mathcal{T}_d) < \infty$) is required* *for time-critical applications. Therefore, a* light-tailed *distribution of $\mathcal{T}_d$ is assumed or required in many deployments and performance studies of wireless networks in the literature. For example, the authors in [57] implicitly assume that the dissemination latency is exponentially bounded (light-tailed) so as to make their delay-capacity tradeoff analysis tractable.*

Theorem 17 reveals that to achieve a light-tailed *dissemination latency (note that* Gamma distribution *is a type of light-tailed distribution), the* mobility capability *of secondary users α need to be larger than some critical value, which will be specifically identified in Propositions for* **EIHP**, **PIHP** *and* **HHP** *mobility, respectively [68]. This result encourages and validates the existing endeavor of deploying CRN for practical applications, including time-critical applications, such as emergency networks and military networks. In addition, the result in Theorem 17 also motivates further performance studies of CRNs, for example, the delay-capacity tradeoff study.*

We must emphasize that the goal of this study is to investigate the fundamental properties of the dissemination latency $\mathcal{T}_d$ in CRNs where secondary users are mobile according to the **general** *mobility patterns. However, if given more knowledge of the CRN, e.g.,the specific mobility patterns, the same proof can also be used to derive the more specific distributions of $\mathcal{T}_d$*

As the network size of the CRN increases, we next have the following theorem on the availability of the dissemination latency $\mathcal{T}_d$.

***Theorem* 18.** *We consider two types of connectivity in large CRNs:* full connectivity *and* percolation-based connectivity. *The former is that there exists a communication path between any two nodes; and the latter is that there exist a large component well scattered over the entire network. CRN $(\mathcal{F}_{m,n}, \mathcal{M}(\Phi, \Psi, \alpha), \mathcal{M}_H)$, there exists a finite constant κ such that $\mathbb{P}(\lim_{\mathcal{D}\to\infty} \mathcal{S}_d = \lim_{\mathcal{D}\to\infty} \frac{\mathcal{T}_d}{\mathcal{D}} = \kappa) = 1$.*

Remark 14. *Scalability has been the most fundamental concern that has so far discouraged the deployment of large wireless networks. Among the several scalability issues, perhaps the most basic one is that related to the dissemination latency. Theorem 18 demonstrates that in large connected CRNs, the dissemination latency $\mathcal{T}_d$ asymptotically scales linearly with the initial "distance" between the source and destination, i.e., the message sent by a source reaches its destination at a fixed asymptotic speed. This result enables and verifies the deployment of CRNs for large applications, such as sensor networks.*

This paper aims to understand the fundamental properties of the dissemination latency $\mathcal{T}_d$ in CRNs under general mobility. However, besides the theoretical importance of our findings, our results can be used practically not only in the initial deployment of a CRN, but also in evaluating the performance of specific CRN applications. For example, in a large deployment of a mobile CRN as a wireless sensor network, the result in Theorem 18 can be used to estimate the delay elapsed between the time at which an incoming event is sensed by some node of the network and the time at which this information is retrieved by the data collecting sink. In the next two sections, we will present the proof for Theorem 17 and Theorem 18, which studies the distribution and scalability of the dissemination latency $\mathcal{T}_d$ in finite and large CRNs under **EIHP**, **PIHP** and **HHP** mobility, respectively.

The Scalability of $\mathcal{T}_d$ in Large CRN

We have studied the distribution of the dissemination latency $\mathcal{T}_d$ in finite mobile CRNs and proved Theorem 17. We next prove Theorem 18, which studies the *scalability* of the dissemination latency $\mathcal{T}_d$ in large mobile

CRNs and states that $\mathcal{T}_d$ asymptotically scales linearly with the "distance" between the source and destination nodes. Particularly, our results demonstrate that as the network size (n and m) increases to ∞, $E(\mathcal{T}_d) \to \infty$ even with large *mobility capability* α. Thus in large CRNs (i.e., the network size is large enough), the distribution of $\mathcal{T}_d$ cannot be used to measure *how fast* information is disseminated. Therefore, we will investigate, instead of the distribution, the *scalability* of $\mathcal{T}_d$ in large CRNs. Specifically, we will study the scaling behavior of $\mathcal{T}_d$ with respect to the "distance" $\mathcal{D}$ between the source v_s and destination v_d, which can be characterized by the "speed" $\mathcal{S}_d = \frac{\mathcal{T}_d}{\mathcal{D}}$.

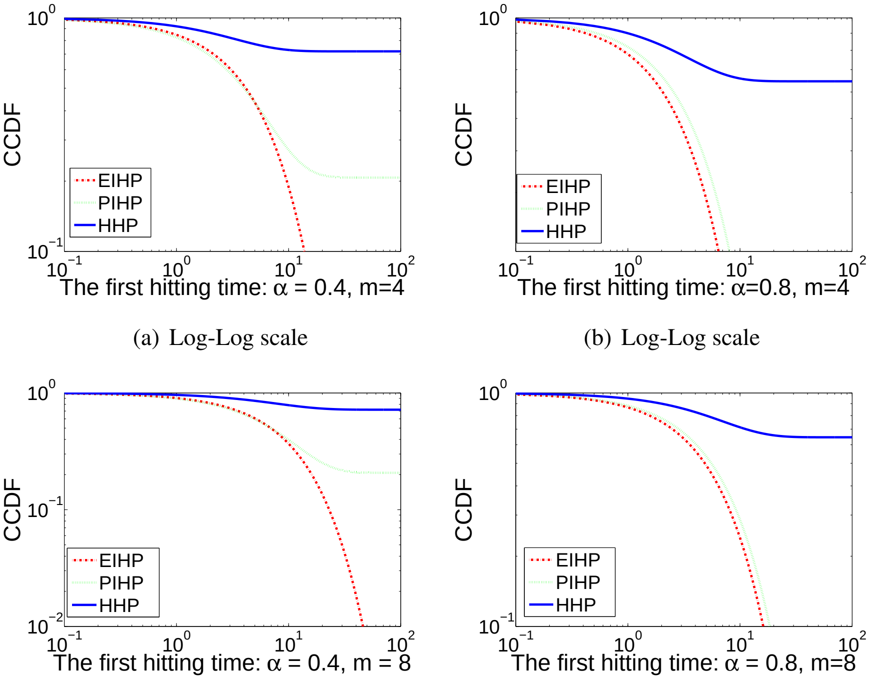
Based on the distribution study of $\mathcal{T}_d$ explained earlier, we have the implication that the tail of $\mathcal{S}_d$ may "disappear" as the network size increases. We rigorously study in this section the scalability of $\mathcal{T}_d$, which validate our implication (see Theorem 18). We must emphasize that our derivation is based on the assumption that the number of nodes n and m are finite. However, n and m may approach to ∞ in large CRNs. Therefore, the techniques and results in finite CRNs may not be applied to large CRNs directly. Instead, we will use *large number theory* to demonstrate that $\mathcal{T}_d$ asymptotically scales linearly with $\mathcal{D}$, i.e., $\lim_{\mathcal{D}\to\infty} \mathcal{S}_d$ is convergent to some positive constant. Specifically, the main tool used is *Liggett's subadditive ergodic theorem, [69].*

Note that Liggett's theorem provides a method to study the *limiting behavior* of a large random process. We next show how to use Liggett's theorem to study the *limit* of the dissemination speed $\mathcal{S}_d$. Before we proceed, we need first to make some clarification about our model $(\mathcal{F}_{m,n}, \mathcal{M}(\Phi, \Psi, \alpha), \mathcal{M}_H)$ for large CRNs. To study the dissemination latency $\mathcal{T}_d$ and speed $\mathcal{S}_d$ in large CRNs, we progressively increase the number of secondary and primary users n and m in Ω_n. Note that as m and n increases, the homogeneously distributed primary users are asymptotically distributed as a two-dimensional Poisson point process with density $\lambda_p = \lambda n/m$. Assume that λ_p is a constant for simplicity (i.e., m and n increases proportionally). Another implicit assumption is that λ_p is not large thus there exist enough spectrum opportunities for secondary communication. This assumption is reasonable since the *low spectrum utilization* of primary users is the motivation for CRNs. And similarly, to facilitate the analysis, we have proved Theorem 18 under three subclasses of mobility **EIHP**, **PIHP** and **HHP** respectively.

4.5.3 Simulation Results

In this section we provide simulation results to support our theoretical analysis on *distribution* and *scalability* of latency in finite and infinite CRNs, respectively. In these simulations, time is partitioned into unit slots and in each time slot, primary users are uniformly distributed at random within the network area and secondary users are uniformly distributed around their home points (i.e., Ψ is uniform). Furthermore, home points are uniformly distributed around the center points under PIHP mobility (i.e., Φ_P is uniform). The transmission range r of secondary users and the interference range R_I of primary users are set as $r = 0.1$ kilometer (km) and $R_I = 0.3$ (km), respectively. Secondary users opportunistically access $m = 2$ channels.

We first study a finite CRN where $n = 16$ secondary users are mobile within an 2×2 (km^2) area (i.e., $\lambda = 4$ per km^2). Figure 35 illustrates the complementary distribution (CCDF) of the dissemination latency $\mathbb{P}(\mathcal{T}_d > t)$ on a log-log scale for EIHP, PIHP and HHP models with different values of the *mobility radius* α and the spatial density λ_p of primary users. The probability is calculated based on the average of 1000 independent



(a) Log-Log scale

(b) Log-Log scale

(c) Log-Log scale

(d) Log-Log scale

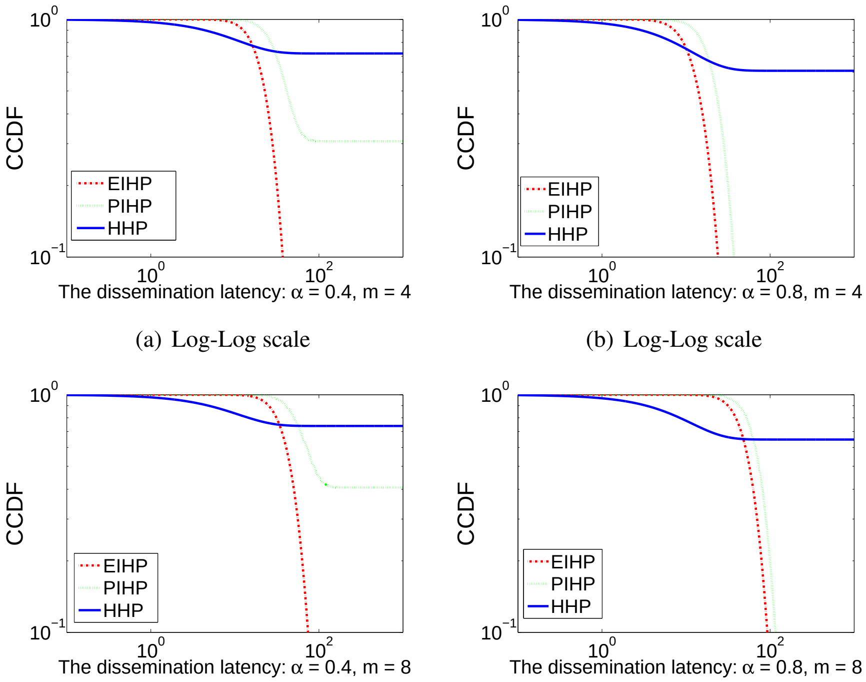
Figure 34: CCDF of the first hitting time $\mathcal{T}_h(v_i, v_j)$ between neighboring secondary users v_i and v_j.

simulations. It is observed in Figure 35 that as λ_p increases, the curves move right-ward, which indicates the increasing expected dissemination latency. However, regardless of the value of λ_p, when $\alpha = 0.4$ (km), which is larger than the *cutoff point* under *EIHP* but smaller than those under PIHP and HHP, the dissemination latency $\mathcal{T}_d$ has a light tail under EIHP but *heavy tails* under PIHP and HHP. As α increases to 0.6 (km), which is larger than the *cutoff point* in PIHP, but still less than that in HHP, the *heavy tail* of $\mathcal{T}_d$ in PIHP disappears, but $\mathcal{T}_d$ in HHP presents a *heavy* tail. These results are in good agreement with our theoretical analysis.

We further perform a series of simulations to validate our asymptotic results in large networks. Figure 36(a) and 36(b) show the latency *scalability* in large CRNs under EIHP and PIHP models, respectively, where the spatial density of secondary users is $\lambda = 4$ (per km^2). As shown in 36(a) and 36(b), no matter how large the *mobility radius* α is, the dissemination latency $\mathcal{T}_d$ scales linearly with the dissemination distance $\mathcal{D}$ (Manhattan distance). Moreover, The *latency scalability* in a large percolated CRN under HHP mobility, where the spatial density of secondary users is set as $\lambda = 200$ (per km^2) to ensure percolation, which shows that in percolated CRNs, the dissemination latency $\mathcal{T}_d$ scales linearly with the dissemination distance $\mathcal{D}$ (Euclidean distance) as $\mathcal{D}$ increases. In addition, as shown in Figure 36, the *scalability* decreases as the spatial density λ_p increases. These observations provide a straightforward illustration of Propositions [68].

4.5.4 Relevance to Original Goals

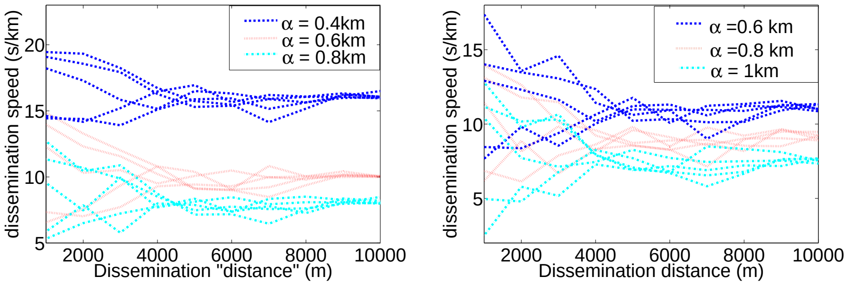
In summary, we study the distribution of the information dissemination latency $\mathcal{T}_d$ in *finite* CRNs and the scalability of $\mathcal{T}_d$ in *large* CRNs under general mobility. We found that in finite networks, there exists a cutoff point



(a) Log-Log scale (b) Log-Log scale

(c) Log-Log scale (d) Log-Log scale

Figure 35: CCDF of the dissemination latency $\mathcal{T}_d$ under general mobility $\mathcal{M}(\Phi_H, \Psi, \alpha)$.



(a) EIHP (b) PIHP

Figure 36: The dissemination speed $\mathcal{S}_d = \frac{\mathcal{T}_d}{\mathcal{D}}$ (s/km) for 5 independent simulations with parameters $\lambda = 4$ and $\lambda_p = 0.5$, respectively.

on the *mobility radius* α of secondary users, above which the tail distribution of $\mathcal{T}_d$ is bounded by some Gamma distribution and below which $\mathcal{T}_d$ has a *heavy-tailed* distribution. When networks become large, the dissemination latency $\mathcal{T}_d$ is (linearly) scalable with respect to the dissemination distance. Our results demonstrate that when secondary users can move in a large region, a *Gamma* distributed (light-tailed) latency in finite networks, or a scalable latency in large networks, is achievable, which encourages the deployment of CRNs for immediate communications in the aftermath of WMD attacks, which implies that inter-operation among different networks would make a big difference in delivery messages and large-scale applications.

5 Dection, Localization, and Tracking of Systematic Failures

Our primary methodology is to use algebraic geometry to identify failures in terms of coverage holes. While it has a wider application scope, we have focused in this work, on sensor networks, for both the sake of concreteness, as well as the for the generic properties of sensor networks which make them very suitable for such basic research into the *impact of weapons of mass destruction on networks*. Sensor networks are an important class of distributed and pervasive systems, with applications in areas including environmental monitoring, health care and military operations [70]. A unifying theme of many of these problems is to glean consensus information by systematically combining the data collected at individual nodes, in accordance to the structure of the network. The consensus information thus obtained characterizes the network, or the data in the network as a whole, and better represents the underlying phenomenon than may be inferred from the data at individual nodes. This reveals the fundamental nature of sensor networks, where global patterns emerge from simple interactions between nodes.

In order to remain robust to unprecedented scenarios and limitations of resources, a likely situation in the event of a WMD attack, we are motivated to rigorously seek, the minimal required information, and tools needed to process this information, so as to perform a certain task. Owing to limited power and communication capabilities, and for robustness, we are motivated to develop distributed algorithms. The information which is readily available in sensor networks, is for each node to know its neighboring nodes. Two nodes are neighboring nodes if they can communicate with each other. This is equivalent to having a distributed representation of the communication graph. With limited communication between the nodes, we may also obtain the higher order cliques in the graph, and the sub-cliques through which they connect to other cliques.

We observe that many tasks in sensor networks may alternatively be stated in topological terms. The tasks of detection and localization of coverage holes and worm holes are two such examples. Coverage holes may also be caused due to a large localized failure in the network, such as that caused by a WMD. The combinatorial information mentioned above is sufficient to compute topological invariants, and is the subject of Algebraic Topology [71]. We employ this theory to develop distributed algorithms to detect and localize coverage and worm holes. We emphasize that this is the first work which simultaneously solves both problems, supporting our thesis that algebraic topology offers a general framework for topological analysis in sensor networks, and other applications in general networks.

Perhaps the more likely scenario produced by a WMD is that of a dynamically varying network produced by spatially and temporally correlated, for potential future mitigation of *systematic* failures. In this context, we investigated the problem of tracking such systematic failures, a process which led us to unify several existing definitions of network boundary in the literature into a single mathematical framework, and develop efficient algorithms to compute and update network boundaries. Tracking such failures provides information about the nature of a failure, helps in estimating its impact in the future and develop counter-measures.

It is often the case that surveillance of a region is performed using mobile agents/robots/UAVs, etc., (which we will model as nodes for simplicity). In such cases, analyzing the network formed by these mobile nodes provides us with valuable information which the nodes do not, on their own. However, the mobile nature of the network imposes difficult challenges in its analysis. We showed that the recently developed theory on zig-zag persistent homology [16] can also be effectively used to analyze mobile networks, and efficiently track which lack coverage, again without the need for any co-ordinate information. These strategies also help analyze and compare mobility patterns which was not possible with previously available techniques.

5.1 Localization of Failures in Large-Scale Networks: A Manifold Learning Approach in Data Space

To assess the health of a network, we may have access to a variety of measurements at the individual nodes. The characterization of these measurements yields a variety of information, which include that about the functionality of the network. To cope with an onset or the aftermath of a significant attack in the likes of one by weapons of mass disruption/destruction, our focus primarily lies in detecting early failures and in localizing them to infer the degree to which the topology of a network is preserved. The power of a detection strategy lies in its versatility and its adaptability to the data $\{x_i\}_{i=1,\cdots,N}$ (N is the number of nodes) which may be available at a given point in time. The rationale of our proposed technique is based on the fact that *normal measurements* of a network *lie on some manifold* to be learned, and any interruption/deviation thereof is an indication of a failure. In what follows we describe our technique which is for any arbitrary network topology and is independent of the data nature measured at the network and which provides the first alert of malfunction. We are currently working on the *localization* of the failure and its direct impact on the overall topology of the network.

There are a variety of manifold learning algorithms, and for simple illustration we choose to use isomap algorithm. This choice is due to the fact that isomap has provided successful embedding results and has its theory has been widely studied [72]. The principle and motivations of this work are, however, extensible to other mapping (embedding) algorithms.

5.1.1 Isometric Feature Mapping (Isomap)

Isomap is a nonlinear mapping algorithm that starts from the assumption that the high dimensional data lie on a Riemannian manifold [73]. To achieve the dimension reduction, isomap defines a mapping that aims to preserve the geodesic distances on the initial manifold. We may describe isomap as merely an improved version of Multidimensional Scaling (MDS) embedding where the interpoint distance is restricted to lie on the initial manifold of the data. Given a data point sample of N points X_i, $i = 1, \cdots, N$, from a d-dimensional manifold $\mathcal{M}$, where $\mathcal{M} \in \mathbb{R}^n$ and $d < n$, we describe the different steps of isomap embedding algorithm in Table 2.

5.1.2 Adaptive Isomap

Although classical isomap shows good embedding results, it remains very unstable and sensitive to noise and to the choice of the parameter ϵ and the distance function used prior to applying MDS. Changing the distance from Euclidean to geodesic appears, thus, to be insufficient to completely respect the intrinsic geometric structure of the initial manifold $\mathcal{M}$.

In what follows we propose to introduce a different distance matrix $\mathbf{D}_M$ to replace $\mathbf{D}_E$ in the algorithm described in Table 2. Our objective is to define, each time, a distance that is fully dependent on the sample point $\{X_i\}_{i=1}^{N}$. We, hence, rescale the data coordinates based on their distributions on $\mathcal{M}$ as well as their correlations.

Table 2: Non-adaptive isomap algorithm.

```
**Step 1:** Construct a weighted graph $G$;

Let $G = \{\mathcal{A}, \mathbf{X}\}$, where:
  • $\mathcal{A}$ is an $(N \times N)$ adjacency matrix;
  • $\mathbf{X} = [X_1, \cdots, X_N]^T$;
Compute $\mathbf{D}_E$, the matrix of Euclidean distances between each two points in $\{X_i\}_{i=1}^N$.
Choose $\epsilon$, the neighborhood radius.
for $i, j \in \{1, \cdots, N\}$ do
     if $\mathbf{D}_E(i, j) < \epsilon$ do
          $\mathcal{A}(i, j) = \mathbf{D}_E(i, j)$;
     else
          $\mathcal{A}(i, j) = \infty$;
     end if
end for

**Step 2:** Compute geodesic distances on $G$.

Let $\mathbf{D}_G$ be the matrix of geodesic distances between each two points in $\{X_i\}_{i=1}^N$.
do $\mathbf{D}_G = \mathcal{A}$; (initialization) for $i, j \in \{1, \cdots, N\}$; $k = 1$; do
     while $\mathbf{D}_G(i, j) \neq \mathbf{D}_G(i, k) + \mathbf{D}_G(k, j)$ do
          for $k \in \{1, \cdots, N\}$ do
            $\mathbf{D}_G(i, j) = \min(\mathbf{D}_G(i, j), \mathbf{D}_G(i, k) + \mathbf{D}_G(k, j))$;
          end for
     end while

**Step 3:** Apply MDS on $\mathbf{D}_G$.
```

Table 3: Description of the learning step (new adaptive distance)

```
**Step 0:** Compute $\mathbf{D}_M$, the new distance on $\mathcal{M}$.

Choose $\epsilon_1$, the neighborhood radius for manifold learning;
and $\epsilon_2$, the neighborhood radius for the construction of $G$.
for $i \in \{1, \cdots, N\}$ do
    $\mathbf{Y}_i = X_i^T$; (initialization)
    for $j = 1, cdots, N$ do
        while $\mathbf{D}_E(i,j) < \epsilon_1$ do
            $\mathbf{Y}_i = [\mathbf{Y}_i; X_j^T]$;
        end while
    end for
    $\sum_i = \mathrm{cov}(\mathbf{Y}_i^T)$, $\mathrm{cov}(\cdot)$ being the covariance matrix;
end for
for $i \in \{1, \cdots, N\}$ do
    for $j \in \{1, \cdots, N\}$ do
        $\mathbf{D}_M(i,j) = (X_j - X_i) \sum_i^{-1} (X_j - X_i)^T$;
    end for
end for
do $\epsilon = \epsilon_2$; $\mathbf{D}_E = \mathbf{D}_M$;
go to Step 1. (See Table 2)
```

Because it relies on a learning procedure, we refer to this modified isomap as adaptive isomap algorithm. We start, therefore, the algorithm of Table 2 with a learning step, *i.e*, Step 0, as described in Table 3.

5.1.3 Performance comparison

In this section, we qualitatively and quantitatively compare the performances the the versions (adaptive and non-adaptive) of isomap embeddings. To that end, we start by defining a performance measure. We, then simulate different classical examples of manifolds to embed in a lower dimensional space. The choice of our examples is such that one may visually inspect and verify the properties and the intuitions behind each technique.

We choose residual variance ρ to be our performance indicator. In further applications, we will show we may use it to investigate the topological structure of a manifold. Residual variance ρ is defined in (58).

$$\rho = 1 - \big(\mathrm{corrcoef}(\mathbf{D}^X, \mathbf{D}^Z)\big)^2, \tag{58}$$

where:

- $\mathbf{D}^X$: is the distance matrix for the initial data in $\mathcal{M}$. For isomap embedding technique, this matrix is the geodesic distance matrix $\mathbf{D}_G$ that is fed into MDS. We note that $\mathbf{D}_G$ changes depending on the first distance matrix used to identify the neighborhood. For the classical non-adaptive isomap, this distance was simply Euclidean, *i.e.*, $\mathbf{D}_E$, while for the adaptive case we defined it as $\mathbf{D}_M$. This difference is crucial in comparing the performances of the two versions of isomap algorithm.

- $\mathbf{D}^Z$: is the distance matrix for the final (embedded) data of reduced dimension p ($p < n$). We take advantage of the simplicity of the geometries of our examples (swiss rolls, hemispheres, parallel sheets), and take p equal to 2 and consider $\mathbf{D}^Z$ to be Euclidean. It becomes trivial to visually verify the accuracy of our assumption. For more complex geometrical and topological structures, one would, however, need to compute geodesic distances on the new manifold embedded in $\mathbb{R}^p$
- corrcoeff$(\cdot,\cdot)$: is the linear correlation coefficient. If we note $\{d_X\}$ and $\{d_Z\}$ as two ordered sets of the distances (matrix elements) of $\mathbf{D}^X$ and $\mathbf{D}^Z$, respectively, then:

$$\text{corrcoef}(\mathbf{D}^X, \mathbf{D}^Z) = \frac{\sigma_{XZ}}{\sigma_X \sigma_Z}, \tag{59}$$

σ_{XZ} being the correlation between the two sets $\{d_X\}$ and $\{d_Z\}$, and σ_X and σ_Z being the standard deviations of $\{d_X\}$ and $\{d_Z\}$, respectively.

As stated earlier, in the presence of noise, non-adaptive isomap sees its performances drastically dropping. We, hence, test how well/bad adaptive isomap performs on the same noisy datasets. We use the data point samples (omitted due to file size). We add Gaussian noise to them with standard deviations varying between 0% and 8% of the orthogonal distance between the two parallel sheets, the normal distance between two consecutive levels of the swiss roll, and the orthogonal distance between the poles of the two adjacent hemispheres.

In summary, a manifold learning approach is explored to identify the locations of attacks or failures by detecting variations in massive data space under the assumption that many sensors (e.g., road-side monitoring sensors) are used to collect information for normal operation of a network. Regardless of the contents being collected by sensors, it is important to localize the incidents by examining data set. As an initial effort, we have demonstrated that manifold learning is an effective approach to reduce the dimension of high-dimensional data set and to detect points of failure with abnormal data.

5.1.4 Coverage hole detection and localization

We consider a problem where a set of sensors, with capability to communicate with neighboring sensors and perform simple arithmetic operations, are deployed in a region for surveillance and monitoring purposes. We say that there is a hole in the coverage if any part of the intended region of surveillance is not in the range of any sensor. This might be due to failure of deployment, or intentional sabotage. One such scenario is shown in Figure 37 Our goal is to detect and locate these holes in the coverage in *a robust, resource efficient, fast, and distributive* manner. The problem is stated more precisely as follows.

Problem Formulation Consider N sensor nodes randomly deployed in a region of interest $\mathcal{R}$ in a plane. We denote the collection of all the nodes as the set $V = \{v_i\}$. Each node v_i can communicate with a set of neighboring nodes $\mathcal{N}_i$, in its vicinity. A communication graph $G = (V, E)$ is thus formed as the collection of the set

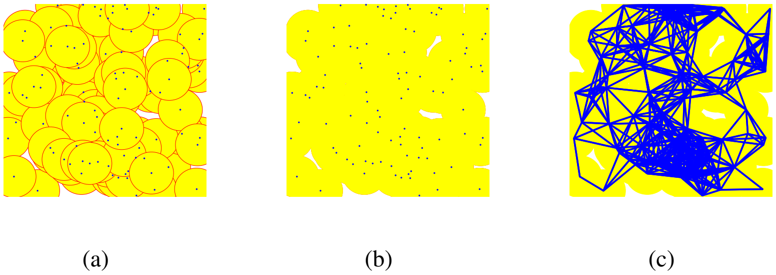


Figure 37: Figure shows the problem context for localizing coverage holes. The union of balls around each sensor (a) is called the coverage ares (b). Holes in the coverage area implies that that region remains unmonitored, and hence indicates a failure in coverage. The only information available is the proximity neighborhood for each node, as summarized by the communication graph (c).

V together with the set of edges $E = \{(v_i, v_j)\}$ where $(v_i, v_j) \in E$, if and only if v_i, v_j can communicate with each other. A graph is said to be a unit disk graph, denoted by $G_1 = (V, E_1)$, when $(v_i, v_j) \in E_1$ if and only if $d(v_i, v_j) \leq 1$. A graph is said to be a $\epsilon-$quasi unit disk graph, denoted by $G_1^\epsilon = (V, E_1^\epsilon)$, if 1) $(v_i, v_j) \in E_1^\epsilon$ whenever $d(v_i, v_j) \leq 1 - \epsilon$, 2) $(v_i, v_j) \in E_1^\epsilon$ with probability 0.5 whenever $1 - \epsilon < d(v_i, v_j) \leq 1$, and 3) $(v_i, v_j) \notin E_1^\epsilon$ whenever $d(v_i, v_j) > 1$. We model the graph as a quasi-unit disk graph due to imperfections in the antenna's directional sensitivity.

Let R_c^i denote the coverage area of the sensor on node v_i, and the union $R_c = \cup_i R_c^i$ is the coverage area of the network. The existence of coverage holes is determined by whether or not the following set relation holds:

$$\mathcal{R} \subseteq R_c = \cup_i R_c^i$$

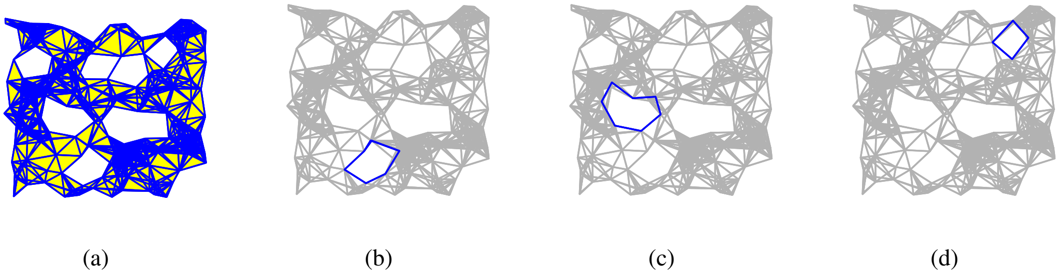


Figure 38: Figure shows the results of the coverage hole localization algorithm. The input to the algorithm is Rips complex (constructed locally from the communication graph) and the result of the localization are cycles around holes as shown in (b), (c), and (d).

To better reflect the challenges faced under a WMD attack scenario and to be faithful to the technical difficulties of deploying large number of sensors, we further impose the following restrictions:

1. There is no central location which can gather all the data from the network for processing.

2. Each sensor only knows the identities of senors in its geographic proximity, and nothing else.

3. The senors are not aware their global location, and there is no means to obtain this information.
4. The reception strength of the antennas is not uniform in all directions owing to imperfections in manufacturing, and more importantly, due to presence of obstacles and fading.

We are happy to report that even under these restrictions, we have developed a provably correct algorithm [74] which outperforms the state of the art algorithms [75] for similar (often with less stringent restrictions) problems.

Results. The output of our algorithm for an example sensor network is shown in Figure 38. The left most figure shows the input network with coverage holes, and the remaining figures shows the locations of the coverage holes as determined by our algorithm. The preliminary versions of these results were published in [76], and the complete work in [74], where we analyze the complexity in detail. To give a sense of the comparison, the complexity of our algorithms is same as that of computing a starting point for the state of art algorithm for solving the same problem of localization under the restriction assumed here.

5.2 Failure Identification and Localization

The infrastructure of computing systems is rapidly transitioning from centralized systems to distributed and pervasive systems. A very important category of such systems are sensor networks which find applications in areas including Environmental monitoring, Health care and Military operations [70]. There has been a considerable research interest in this field over the past decade addressing problems including node localization [77], distributed compression [78], probabilistic inference and motion tracking. A unifying theme of many of these problems is to glean consensus information by systematically combining the data collected at individual nodes in accordance to the structure of the network. The consensus information thus obtained characterizes the network, or the data in the network, as a whole and better represents the underlying phenomenon which can be inferred from the data at individual nodes. This reveals the fundamental nature of sensor networks: they are essentially *complex networks* in which global patterns emerge from simple interactions between nodes. From an engineering perspective, the fundamental challenge in sensor network applications is to cope with the limited resources; a limited communication capability of nodes, i.e. nodes can communicate only with their neighbors, a limited power and a limited memory. Furthermore, sensor networks are often deployed in inaccessible locations and situations where maintenance is impractical; this makes careful use of exhaustible resources such as power, imperative.

This unique combination motivates the use of techniques such as topological analysis, which directly extracts global information without being overly dependent on the local structure and thereby alleviating the need for excessive resources.

5.2.1 Objectives and Approaches

There are two main categories of techniques in literature relating to the analysis of the topology: Morse Theoretic and Algebraic Topological techniques. In our endeavor to analyze the topology of spaces of interest, we predominantly followed the Algebraic Topological methodology. Algebraic topology, in contrast to a Morse theoretic approach, is relatively a more direct technique to analyzing the topology of a space easily expressed in terms of algebraic objects. There is an extensive literature in Algebraic topology [79, 80] which shows a very strong relationship between topological spaces and their algebraic counterparts. Also, this enables us to draw from the extensive pool of knowledge from algebra to develop fast and efficient algorithms. The assigned algebraic objects have the following important properties;

- They directly reflect the topological features of an underlying space.
- They are invariant to continuous deformations.

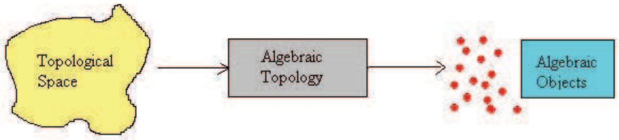


Figure 39: A high level schematic of Algebraic Topology.

The use of algebraic topology for the coverage problem was introduced in [81, 82, 83] demonstrates distributed computation of homology groups and [84] attempts to localize the holes by posing the localization as an optimization problem. We further exploit the spatially constrained nature of the coverage holes and formulate a very effective "divide and conquer" algorithm.

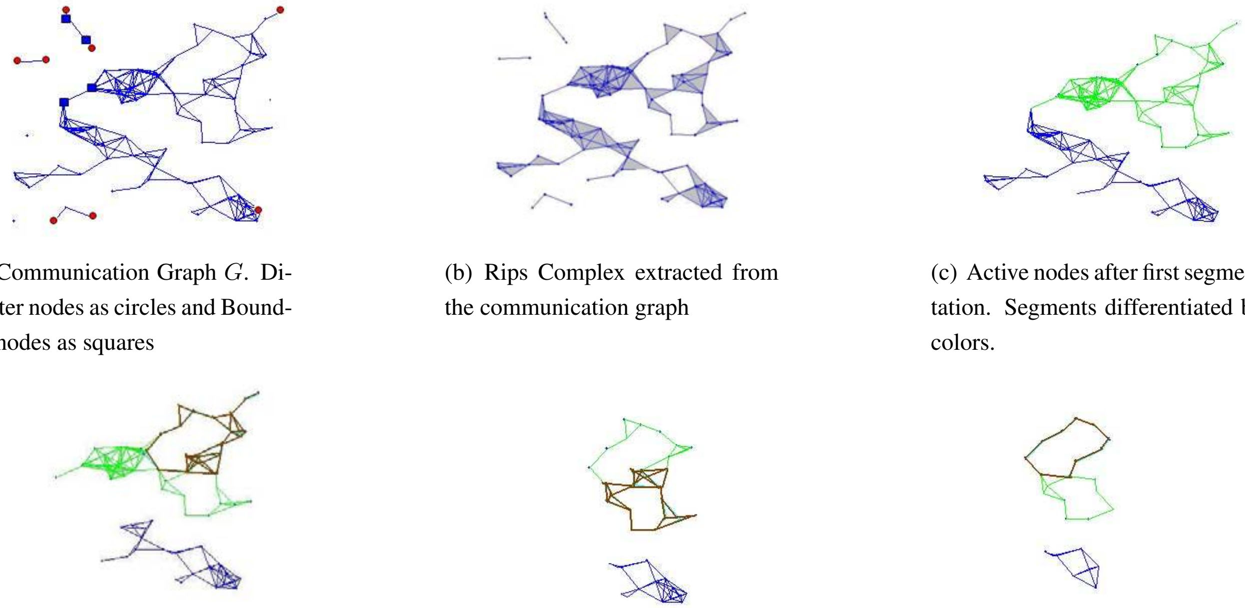
5.2.2 Localization of Failures

Main Results: We demonstrate the merits of such analysis by exploiting tools to solve two specific important problems:

- A Coverage Hole detection and localization
- A Worm-Hole Attack detection and localization

The problem of coverage hole localization seeks to identify a closed contour in the network which encloses a specific region of interest. This region can be the part of the sensor network which has been destroyed by a WMD attack or a region indicating some other undesirable parameters, for example, radiation level above a given threshold. We further demonstrate the effectiveness of such topological analysis by looking at more complex attack in the network which, fundamentally, alters the perceived topology of the network in order to significantly disrupt its usage.

Given a set of sensors monitoring a given region of interest, with each sensor having a certain coverage area, we want to confirm whether each point in this region is covered by at least one sensor. Further, if this is not the case, we want to find the set of sensors surrounding the region which is not covered (coverage hole).



(a) Communication Graph G. Diameter nodes as circles and Boundary nodes as squares

(b) Rips Complex extracted from the communication graph

(c) Active nodes after first segmentation. Segments differentiated by colors.

(d) Active nodes after two iterations

(e) Active nodes after three iterations

(f) Active nodes after four iterations

Figure 40: Simulations showing convergence to hole locations

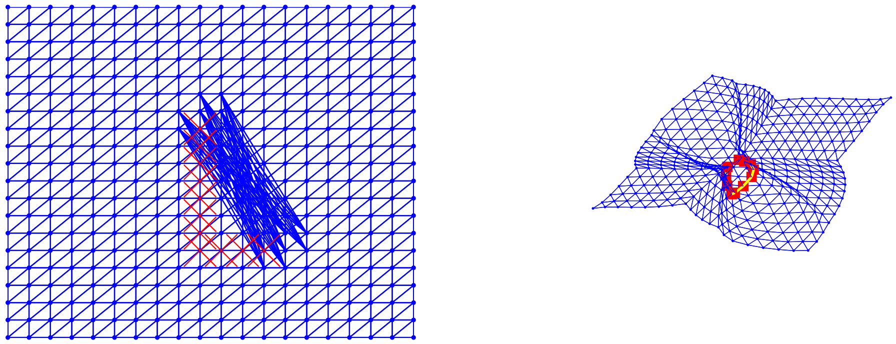
A computer simulation of the proposed algorithm is shown in Figure 40.

Problem Statement: We wish to first detect if there is a worm hole attack taking place in the network. Further, if an attack is detected, we wish to locate precisely the sensor nodes surrounding the attack nodes so as to quarantine them and nullify the attack.

5.2.3 Worm hole detection and localization

A worm hole attack is typically launched by two colluding external attackers who do not authenticate themselves as legitimate nodes to the network. When initiating a wormhole attack, an attacker overhears packets in one part of the network, tunnels them through the wormhole link (external to the network) to another part of the network. This effectively generates a false scenario of the presence of the original sender in the neighborhood of the remote location. Many routing algorithms depend on the nodes' ability to accurately discover their neighboring nodes. The nodes ordinarily perform a broadcasting beacon (including ID, and other information) to their neighbors. If the neighbor discovery beacons are tunneled through wormholes, the good nodes will get false information about their route. Although finding faulty routes is in itself a problem, worm holes can cause further critical security threats using these faulty routes. The resulting effect of wormholes on the routing, is to include a worm hole link in most of the computed routes. This in turn, gives an attacker complete control of transmitting great amounts of data, which may be selectively or completely dropped.

Figure 41 demonstrates this change in topology for a grid network. Many important functions such as routing

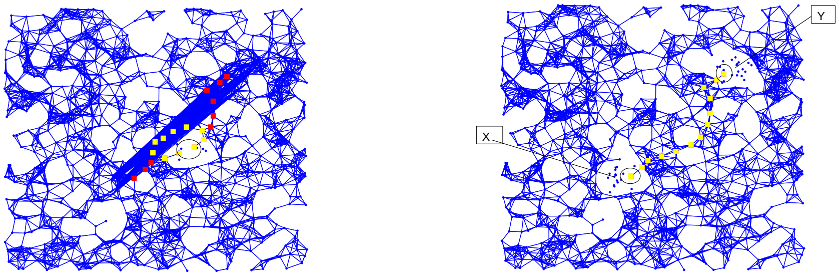


(a) network grid with links caused because of a worm-hole

(b) the same grid shown in 3d to respect distant properties measured as hop distances

Figure 41: Deformation in network structure because of a worm hole. The cycle created is shown in red

and localization [85, 86] can be severely affected by these attacks.



(a) v_1, v_2 not in vicinity of X and Y. A shortest path can be found in the network surrounding the nodes removed.

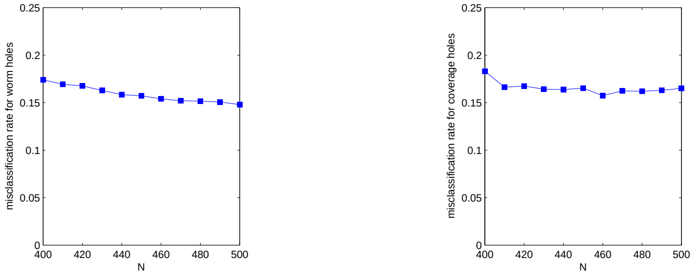
(b) v_1, v_2 in vicinity of X and Y. Alternative shortest path includes all the nodes in the cycle

Figure 42: Worm hole localization algorithm. When an edge is removed from the shortest cycle, and an alternate path is sought between the incident nodes, this path consists of all other edges in the shortest cycle (b) only when the edge removed is in the vicinity of the wormhole attack position.

Assume that a worm-hole attack is launched by two colluding nodes at positions p_1 and p_2 inside a network. Denote the neighborhood regions around these points by N_1 and N_2. The two attacking nodes may receive all the packets transmitted from within their respective neighborhoods, and relay them to the other. Denote by V_1 and V_2 the sets of vertices (sensor nodes) which lie in N_1 and N_2 respectively. The result of a worm-hole attack will be to produce a complete bi-partite graph with V_1 and V_2 as the two classes of vertices. The problem of localizing a worm hole attack, hence reduces to identifying the sets V_1 and V_2. Figure 42 shows an example of a worm hole attack. In this case, X and Y are the positions p_1 and p_2 and the neighborhoods A and B are N_1 and N_2 according to our definition.

By a simple observation, we show that the algorithm to find a coverage hole, may be extended to address this problem. We note here that in order for the proposed hole localization algorithm to work in the presence of a wormhole attack, we do not require the attacking nodes to perform any computation, or follow any protocol. We assume that the attacking nodes “carry out their tasks for coverage hole detection ”, i.e., relay the broadcast signals from one position to another. This effectively creates a virtual link in the network, and the nodes in the network perform all the computations. An additional assumption here, is that the behavior of the attacking nodes does not change during the run time of our algorithm. Figure 42 demonstrates the workings of the algorithm, where it is successful in correctly identifying the worm hole location. These results are widely publicly disseminated in [87].

For the sake of clarity, we note that the worm hole localization algorithm does not share the mathematical rigor of its coverage hole counterpart, but remains a heuristics. In the dissertation [87], we also provide an example where it is impossible to distinguish between coverage holes and worm holes in the absence of geometric information. Owing to this lack of rigor, we analyze the empirical performance of algorithm in distinguishing between a coverage and a worm hole as shows in Figure 43. As seen, the algorithm has a good detection and false alarm rates.



(a) misclassification rate for worm holes

(b) misclassification rate for coverage holes

Figure 43: Experimental results show that the algorithm is able to correctly identify both the coverage holes and worm holes about 85% of the time.

5.2.4 Summary and Benefits

As one of the most important objectives of this project, we aim to develop new models and approaches to identify the impact of multiple failures. To this end, we have achieved the following milestones.

- Demonstrated that this is fundamentally a topological problem and formulated the problem in a generalized topological framework. This involves generalizing the notion of a graph into a simplicial complex.
- We used tools from Algebraic Topology to effectively solve the problem.
- We introduced a novel distributed “divide and conquer” algorithm which converges to desired solution significantly faster than the previously proposed methods in the literature.

- We also achieve huge saving in the communication energy required by purring half the active nodes to rest in each iteration.
- We solve the problem in the same generalized framework developed to solve the coverage hole problem thus demonstrating the effectiveness of our framework for a myriad of topological problems in sensor networks.
- We developed a distributed, efficient and fast algorithm to detect the attack and localize the attack position.

5.3 Detection and Tracking of Random Failures

Due to ease of availability and versatility of applications, sensor networks have received much attention in the literature in past decade. Owing to their low cost and pervasive characteristics, sensor networks are ideal to be deployed in, and to monitor hazardous environments such as volcanic eruptions, wild fires, land slides, war zones etc. One feature common among these environments is that they produce events which cause correlated failures in both space and time. Being able to detect and track such failures becomes crucial both for the sake of tracking itself and for any emergency response thereafter. Further, any such tracking algorithm should be simple and very fast so as to decrease the response time. In this paper, we present a simple algorithm which meets the above needs.

In the deployment, works in [88, 89], the authors present a new fault tolerance metric for deployment called the Region based connectivity. For usual k-connectivity criteria, the network is designed so as to ensure the connectivity of the network even when $k-1$ nodes fail, but the nodes may be taken from any part of the network. Region based k-connectivity imposes an additional condition that all these $k-1$ nodes are within a closed region. These type of failures depict, for example, the scenario of WMD attacks. They show that the design using this criteria requires fewer nodes than required for the standard k-connectivity.

In [90], the author presents a statistical test to identify whether a given set of node positions is likely to be produced by a random deployment of nodes. If the given set of node positions does not fall in the confidence region, a systematic failure is detected. The author in [90] assumes the availability of localization information so as compute the voronoi diagram for the given points. Further, he also assumes that deployment is in a convex region. We do not make both the above assumptions.

For topological analysis, the collection of work in this topic deals with detecting failure event region and analysis of its topological changes. In [91, 92, 93], the authors present some in network heuristics to detect a hole formation and collapse, and other topological changes. Further work in [94, 95] formulate and solve the above problem in a concrete mathematical setting using low complexity distributed algorithms.

In contrast to the above mentioned articles: 1)our work deals with catastrophic situations where the nodes inside the region of interest fail completely or unable to communicate with any other node. 2)such nodes cannot participate in response mechanisms which makes the problem more complicated as we can only use the information available in the neighboring nodes to perform any task, 3) Our formulation does not assume any hardware with capabilities to sense the phenomenon causing the failure, and 4) is robust to random failures of nodes. The work in [96] presents a distributed algorithm to track dynamic boundaries but assumes co-ordinate information and no node failures.

Dynamic curve tracking has been extensively studied in image processing, a domain from our perspective, deals with centralized processing with complete co-ordinate information. Significant work in this area can be grouped as 1)active contour analysis with linear/non-linear evolution models using parametric/non-parametric kalman filtering techniques, and 2)variational methods [97, 98] where the curve/boundary of interest is obtained as a solution to a certain functional optimization. Application range widely including land slide tracking[99], volcanic material flow analysis, object tracking [100] etc. These methods cannot be directly adopted here as they assume co-ordinate information and centralized processing.

5.3.1 Objectives and Approaches

We aim to develop a low complexity distributed algorithm for detecting and tracking such failures. We assume that nodes inside the failure region are either destroyed or unable to communicate with any other node. The algorithm presented here does not assume any co-ordinate information for the nodes. We evaluate the algorithm using simulations.

We consider a region $\mathbf{R}$ in which a set of nodes V are deployed. For a communication radius r_c, a communication graph $G_{r_c} = (V, E)$ is induced on V where $(v_1, v_2) \in E \iff d_{12} = |(v_1, v_2)| < r_c$. An edge (v_1, v_2) in G_{r_c} implies that v_1 is within the communication region (and vice-versa) of v_2, or in other words, can be *observed* by v_2. In what follows, we use v_i to denote the node with index i and also its position in $\mathbf{R}$.

Definition 14. *For a time evolving C^2 smooth, simple and closed curve $\mathbf{C}(s,t) : S \times \mathbb{R} \to \mathbf{R}$, a* systematic failure *is defined to be occurring if*

- $\forall t > 0, v_i \in \mathbf{C}(s,t) \Rightarrow v_i$ *fails at time t.*
- *Let $F(t)$ denote the set of nodes failed at or before time t and in $Int\,(\mathbf{C}(s,t))$, then $|F(t)| \geq 2$.*

The time evolution of the curve $\mathbf{C}(s,t)$ can be specified by assigning a velocity vector at each point of the curve in a normal direction. We consider the vector only in the normal direction as any horizontal component will only result in a reparameterization of the curve.

5.3.2 Tracking Random Failures using A Graphic Tool

We use the representation on the graph in which tracking problem is generally formulated as estimation of both $\mathbf{C}(s,t)$ and $\nu(s,t)$. The accurate estimation of both these quantities is not possible in our scenario, because of lack of co-ordinates. Tracking can only be performed from the reference of the graph itself, i.e., the position of $\mathbf{C}(s,t)$ can be given by specifying the nodes in the graph which are "close" to it. But the graph doesn't provide the means to accurately determine $\mathbf{C}(s,t)$. For example, Figure **??** shows a snap shot of a failure event, and the failure region is indicated by the ellipse. As can be seen in this example, a good part of the curve does not have any nodes (which have not yet failed) close to it. Owing to these inherent problems, we formulate it as following:

for each node v_i and a given time t, we estimate the time $\hat{T}_i(t)$ after which the node fails. Specifying the time remaining at each node, captures both the distance to phenomenon causing the failure and the speed at which it is approaching, and also gives us a measure of risk at each node.

Main Results: We describe the tracking algorithm which computes and maintains the estimates of time to failure at each node. As we assume that the sensors do not necessarily have any hardware equipped to sense the phenomenon causing the failure, the only observations that can be made are nodes failing, and the time a node fails is our event of interest. When a node fails, its neighbors make a decision whether the failure represents a systematic failure or a random failure. This decision process will be described. If a systematic failure is detected, the neighbors estimate the speed, with which this failure is happening (locally) and propagate this information into the network. Given this speed, we describe later in this part, a methodology to be used to determine this speed in other parts of the network. We adopt a simple prediction rule, i.e., the speed of approach at any node will remain constant during the time interval between two failures.

Our simulations show that this is a good approximation and we leave the more detailed modeling of this evolution as a part of our future work. As such, we adopt here a predict-and-update methodology used in a Kalman filter. We cannot use a Kalman filter directly as the the evolution statistics are difficult to obtain which is discussed in more detail next. In what follows, we denote the time between any two node contiguous failures as Δt and the difference in failure times of nodes v_i and v_j as Δt_{ij}. Each node v_i, maintains an estimate of time to failure $\hat{T}(t)$, and a speed of approach $\hat{s}_i(t)$. When the node v_i fails at time t, it computes an observed speed $s_i^o(t)$ and uses this observation to compute its estimate $\hat{s}_i(t)$. Note that here and in what follows, we adopt for convenience the notation "a failed node computes", while these computations are actually being performed in the neighboring nodes.

Failed nodes Given the time difference between two node failures, the primary difficulty in estimating the average speed is that the distance between these two nodes is computed using the graph distance (path length) which is only approximate. The work in literature presents important statistical results on random graphs in limiting cases ($n \to \infty$), but little is known about the statistics of path lengths in random graphs with relatively small number of nodes. As such, we compute speeds by considering the time difference only between adjacent nodes. More importantly, we have to wait for a node to fail to make a computation, as this is the only way to measure the time the failure phenomenon takes to reach that node.

When a node v_i fails at time t, the apriori estimate $\hat{s}_i^-(t)$ is computed using a simple prediction model as given in Equation 60. In the event when there are two other neighbors v_j, v_k which failed previously, such that (v_i, v_j, v_k) forms a clique (simulations show that this is very likely to happen), we can compute the speed of approach with reasonable accuracy. Figure 5.3.2 shows the construction for this computation. The red arcs show the part of the evolving curve passing through the nodes. If (v_i, v_j, v_k) form a clique, we can compute the angle θ as we know all the sides of the triangle. The chord v_jX_1 can be approximated as being perpendicular to ov_i; let $r' = |pX_1|$, then

$$\begin{aligned}\frac{r'}{d_i} &= \frac{r}{d_i}\left(1 - cos(\phi)\right) = \frac{r}{d_i}\left(1 - \sqrt{1 - \frac{a^2 sin(\theta_1)}{r^2}}\right)\\ &\leq \frac{r}{d_i}\left(1 - \sqrt{1 - \frac{r_c^2 sin(\theta_1)}{r^2}}\right)\end{aligned}$$

the second step is obtained using the application of sine law on the triangle ov_jv_i. This immediately shows that as $r \to \infty$, $r'/d \to 0$ and therefore, for large r, angle $\angle v_iX_1v_j$ is very close to $\pi/2$. Approximating both the chords v_jX_1 and v_kX_2 as being perpendicular to the line ov_i, we have the following equations:

$$|v_iX_1| = d_i = a\cos(\theta_1), |v_iX_2| = d_i' = b\cos(\theta_2)$$
$$\theta_1 + \theta_2 = \theta, \frac{d_i}{\Delta t_{ij}} = \frac{d_i'}{\Delta t_{ik}}$$

the last equations arises from the fact that the speed is assumed to be constant during the interval Δt_{ij}. Solving, we have

$$d_i = \frac{\sin(\theta)}{\sqrt{\frac{1}{a^2} + \frac{c^2}{b^2} - \frac{2c}{ab}\cos(\theta)}}, c = \frac{d_i'}{d_i},$$

and the speed $s_i^o(t)$ is obtained as being equal to $d_i/\Delta t_{ij}$. As we have the complete information to compute the speed, the estimate at time t will be equal to s_i^o as given in Equation 61. In case we do not have a triangle of failed nodes as assumed, we use Equation 62. The lengths of the edges used in the computation are always greater than the closest distance between the node and the curve. As a result, we end up overestimating the speed. The $\min$ in Equation 62 serves to compensate this effect, and further, since it might not be very accurate, we use a simple smoothing function as given in Equation 63 for updating the estimate.

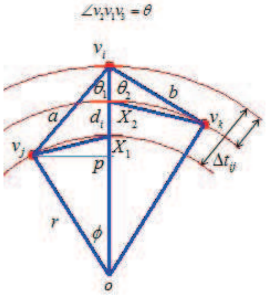


Figure 44: The construction used to compute speed when a node fails.

$$\hat{s}_i^-(t) = \hat{s}_i(t - \Delta t) \tag{60}$$

$$\hat{s}_i(t) = s_i^o(t) \tag{61}$$

$$s_i^o(t) = \min_{j \in N(i)} \frac{d_{ij}}{\Delta t_{ij}} \tag{62}$$

$$\hat{s}_i(t) = \frac{1}{2}\left(\hat{s}_i^-(t) + s_i^o(t)\right). \tag{63}$$

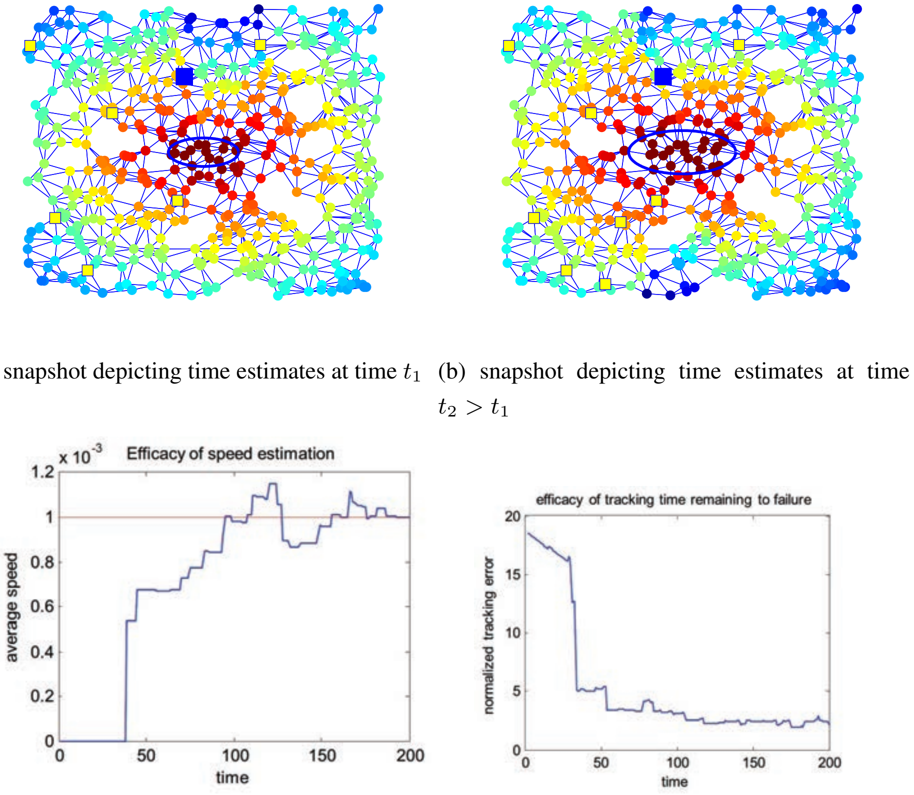
Alive nodes: have not yet failed, the speed prediction is same as in Equation (60). To calculate the $\hat{T}_i$ and update $\hat{s}_i$, each node maintains a distance $\hat{d}_{ic}(t) = \hat{d}_{ic}(t - \Delta t) - \Delta t \hat{s}_i(t - \Delta t)$ which is the estimate of the minimum distance from the node to the curve. When a node v_j fails, its speed estimation is applicable to node v_i (not failed) only when v_j is the closest amongst the failed nodes to node v_i. Figure 70 illustrates this concept. Since the curve evolution is given by the speed in the normal direction, the time which the ellipse (failure phenomenon) takes to reach v is independent of ν_2, and ν_1 is determined by the looking at the point on the curve closest to v. A similar idea can be extended onto the graph. Figure (a) shows all the nodes alive in red, and failed node (shown as blue circle). When the failed node satisfies this condition, we use Equation 64 to update the estimated speed and $\hat{T}_i(t) = \hat{d}_{ic}/\hat{s}_i(t)$.

$$\hat{s}_i(t) = \frac{1}{2}\left(\hat{s}_i^-(t) + s_j^o(t)\right). \tag{64}$$

Random Failures: In order to make our algorithm robust to random failures due to causes other than systematic time evolving failure, we first *detect* the occurrence of a systematic failure before estimating times to failure at the nodes. For this, the following simple rules should suffice: 1)if all the neighbors of a failed node are alive, no speed is estimated and no action is taken, 2)if for a failed node, there is a neighbor which already failed, then a tentative speed is computed and 3) if for a failed node, there is a neighbor which previously failed and has a speed computed, the new failed node computes the speeds and compares it with the tentative speed of its neighbor. If the speeds are close enough, a systematic failure is detected and the information is broadcast to all alive nodes to compute the time to failure. If there are more than one failed neighbors, then a systematic failure is detected if at least one neighbors' speed is compared and confirmed.

The only step which needs some explanation for its implementation is the broadcasting of the information that a systematic failure is detected. When a node fails and a speed is confirmed, its broadcasts this speed to all its neighbors. As the data packets are passed along, they aggregate the edge lengths along their way. Any node broadcasts the information about a node failure only once. Further, if the distance from the failed node v_j to the node v_i is greater than d_{ic}, then v_i does not propagate this information. This way, the information about the node v_j being failed will reach only those nodes for which v_j is closest amongst all the other failed nodes. Based our our description of active nodes, this is sufficient and a significant cost in terms of power is saved.

Simulation and Evaluation: We simulated a time evolving failure on a network with 500 nodes. The time evolving curve was taken to be an ellipse with speed at the tip of the major axis being twice as that at the tip of minor axis and continuously changing in between. Figures 45(a) and 45(b) pictorially depict the time estimates at the nodes for two different times. The red color implies that the nodes estimate relatively smaller time to failure and the blue implies longer times. Figures 45(c) and 45(d) show the error between the speed and time estimates



(a) snapshot depicting time estimates at time t_1 (b) snapshot depicting time estimates at time $t_2 > t_1$

(c) convergence of speed to the true value (d) error in Time estimates vs time

Figure 45: Simulation and evaluation results.

and the true values as a function of time. The results in Figures 45(c) and 45(d) are for a isotropic case where the curve evolving was a circle and speed is constant in all directions. We chose this scenario to illustrate the convergence of speed estimates. The error in speed values converges to zero as time progresses. The error in time estimates converges to a positive value. This is because of the error in approximating the true straight line Euclidean distances between points with the path distance on the graph.

5.3.3 Summary and Discussions:

As one of the most important objectives of this project, we aim to develop new models and approaches to detect and track multiple failures. To this end, we have presented a simple, low complexity and distributed algorithm to detect and track a systematic and time-evolving failure. The accuracy of the results despite a very simple model shows much promise in this strategy. In our future research, we will investigate more sophisticated models for tracking the curve evolution to further improve the accuracy and convergence rates. A significant hurdle to cross would be obtaining good statistics for the error between actual distance between two nodes and the path length on a geometric graph.

In our DTRA funded work to-date, we have developed very efficient, very rapid and novel algorithms to detect and localize static failures viewed as coverage holes, and where the connectivity was primarily defined in terms of range proximity. These algorithms also enjoyed a very important characteristic of being distributed and coordinate-free. The latter feature is particularly important in a WMD environment, where it is more likely that nodes would not be benefiting of Global Positioning System (GPS) help due to its unavailability. A natural follow-up question is then of characterizing the correlated failures caused by one or more WMD events, and distinguishing them from failures which are random and unrelated by using failure detection and tracking algorithms.

5.4 Identification and Tracking of Systematic Failures

Our goal is to detect and track "systematic" failures, which amount to being spatially and temporally correlated. In a region where a sensor network is deployed, catastrophic events such as wild fires, volcanic eruptions or massive explosions, cause large number of node failures. The nature of these events is such that, the positions of the failed nodes are confined to a geographic region. Also, *all* the nodes within this region fail. The area of such region increases with time, and the evolution of its boundary is continuous in time.

5.4.1 Network Classification

The application of network synthesis and classification requires a proper model capturing the properties that control the network evolving as well as the features that characterize the network. To this end, we propose a generalized Markov Graph model, whose dependence structure of the network is used to control the network evolving, and probability distribution of the networks provides features characterizing the networks.

To date, research in network analysis has evolved in many dimensions. For network synthesis, Barabási-Albert model[101] successfully generates networks that follow a power law degree distribution based on the assumption of preferential attachment. In statistics, the Markov Graph model[102], the $p*$ model[103] as well as the models from Statistical Mechanics[104] are well known for characterizing the probability distribution of networks. More recently, we proposed a method based on a Markov Graph model to address the problem of network classification[105].

We were the first to propose a method based on Markov Graph model to classify different types of networks[105]. It makes use of the two features, degree distribution and the number of triads, which determine the probability distribution of networks in Markov Graph model as the crucial features for classification of social networks. In the research area of social network synthesis, Barabási-Albert model as well as many other related models enable the synthesis of the evolution of simple networks whose degree distributions obey the power law[101]. The algorithms for these models are clear and efficient. The open question, however, and in light of these models' shortfall of fully capturing the probabilistic structure of a network, is about the statistical behavior of the required additional features, namely the clustering and crowding coefficients[3]. To this end, we propose a model which provides answers to this question and on both counts.

Dependence structure of the network. In a generalized Markov Graph model, pairs of nodes and triplets of nodes are both considered as basic units in the dependence graph D. We use the term "doublet-node-vertex", denoted

[3] Crowding coefficient is a new feature for networks.

by $\{i,j\}$, to represent the basic unit in D corresponding to a pair of nodes i and j in G and ”triplet-node-vertex”, denoted by $\{a,b,c\}$, for that corresponding to a triplet of nodes a, b and c in G. And $M_{\{a,b\}}$ is the value of the element $\{a,b\}$ in the adjacency matrix $\{M\}$, $T_{\{a,b,c\}}$ represents the state of a triplet-node-vertex, and $T_{\{a,b,c\}} = 1$ when $M_{\{a,b\}}$, $M_{\{b,c\}}$ and $M_{\{c,a\}}$ all take value 1, and $T_{\{a,b,c\}} = 0$ otherwise.

The dependence graph for networks under our proposed generalized Markov Graph model is defined as:

$$D_{GM} = \{node_{DGM}, edge_{DGM}\}.$$

The node set of D_{GM} is defined as $node_{DGM} = N_2 \cup N_3$, where $N_2 = \bigcup_{i,j,i\neq j}\{\{i,j\}\}$ is the set of all doublet-node-vertices, and $N_3 = \bigcup_{i,j,k,i\neq j\neq k}\{\{i,j,k\}\}$ is the set of all triplet-node-vertices. The set $edge_{DGM}$ includes all edges between two nodes in D_{GM} whose states conditionally depend on each other and such a dependence structure is shown as follows:

1. if $\{i,j\} \cap \{a,b\} \neq \varnothing$ and $\{i,j\} \neq \{a,b\}$, the probability mass function $P(M_{\{i,j\}})$ depends on variable $M_{\{a,b\}}$;
2. if $\{i,j\} \cap \{a,b,c\} \neq \varnothing$ and $\{i,j\} \notin \{a,b,c\}$, $P(M_{\{i,j\}})$ depends on variable $T_{\{a,b,c\}}$ and $P(T_{\{a,b,c\}})$ depends on variable $M_{\{i,j\}}$;
3. if $\{i,j,k\} \cap \{a,b,c\} \neq \varnothing$ and $\{i,j,k\} \neq \{a,b,c\}$, $P(T_{\{i,j,k\}})$ depends on variable $T_{\{a,b,c\}}$.

Crucial features for networks. Any pseudo-homogeneous simple network graph G with an associated dependence graph D_{GM}, has a probability mass function:

$$\begin{aligned} P_{GM}(G) =& z^{-1}exp(\sum_k d_k(G)\Theta_k + \sum_c b_c(G)\Gamma_c \\ &+ \sum_{\overrightarrow{m}} t_{\overrightarrow{m}}(G)\tau_{\overrightarrow{m}}), \end{aligned} \tag{65}$$

where $d_k(G)$ is the number of nodes which share the same degree k in network G, Θ_k is the associated coefficient, $b_c(G)$ is the number of nodes which share the same clustering coefficient c, Γ_c is the associated coefficient, $t_{\overrightarrow{m}}(G)$ represents the number of triads which share the same crowding coefficient $\overrightarrow{m}$, and $\tau_{\overrightarrow{m}}$ is the associated coefficient.

The sequence $\{d_k(G)\}$, $\{b_c(G)\}$, $\{t_{\overrightarrow{m}}(G)\}$ correspond to the histograms of the degree list, the clustering coefficient list and the crowding coefficient list, respectively, and their information could be expressed by the degree distribution, the clustering coefficient distribution and the crowding coefficient distribution of network G.

Network synthesis. We evaluate the synthesis power of the generalized Markov Graph model by designing a social network evolving algorithm based on its properties introduced in previous section. According to its first two properties, the probability function for the state of a doublet-node-vertex $\{i,j\}$ in a network should only depend on the state of all the doublet-node-vertices that include node i or node j as well as triplet-node-vertices that include either one:

$$P_{GM}(M_{\{i,j\}}) = f(M_{\{i,1\}}, M_{\{i,2\}}, ..., M_{\{1,j\}}, M_{\{2,j\}} ..., \\ T_{\{i,1,2\}}, T_{\{i,1,3\}}, ..., T_{\{1,2,j\}}, T_{\{1,3,j\}}, ...),$$

where $f(...)$ is a function that depends on the variables $M_{\{i,1\}}$, $M_{\{i,2\}}$, ..., $M_{\{1,j\}}$, $M_{\{2,j\}}$, ..., $T_{\{i,1,2\}}$, $T_{\{i,1,3\}}$, ..., $T_{\{1,2,j\}}$, $T_{\{1,3,j\}}$, ... As an application example of the generalized Markov Graph to network synthesis, we change the preferential attachment algorithm in Barabási-Albert model, which is a special case of a Markov Graph model, to a new algorithm based on a generalized Markov Graph model, by specifying the probability function of the state of a doublet-node-vertex $\{i, j\}$ in network G to be:

$$P_G^{'}(M_{i,j}|i > j) = C\Sigma_{k,k\neq i}M_{\{j,k\}} \\ \Sigma_{k,n,k\neq i,n\neq i}M_{\{j,k\}}M_{\{j,n\}}M_{\{n,k\}}/2,$$

where $\Sigma_{k,k\neq i}M_{\{j,k\}}$ represents the degree of node j, $\Sigma_{k,n,k\neq i,n\neq i}M_{\{j,k\}}M_{\{j,n\}}M_{\{n,k\}}/2$ is the number of triads that include node j, C is a normalization factor, and $i > j$ represents i is a newer node than j. It indicates that the probability is proportional to degree of node j, and to the number of triads that include it. Note that $P_G^{'}(M_{i,j}|i > j)$ is a special case of $P_{GM}(M_{\{i,j\}})$.

Based on $P_G^{'}(M_{i,j}|i > j)$ resulting from a generalized Markov Graph model, we specify the rules of a network evolution and verify that the resulting network satisfies the features described earlier. The rules are as follows:

1. There are m_0 nodes in the network at time $t = 0$.
2. At each time step, a new node with m_0 attached edges is introduced to the network.
3. Each of these edges will find another node as its vertex in the network. The probability of a node being picked is $P_G^{'}(M_{i,j}|i > j)$, where i represents the newly added node, and j for the node being picked.

5.4.2 Systematic Failure Analysis

Sensor networks are often deployed to monitor hazardous environments. Several events/phenomenon in such environments may cause spatially and temporally correlated failures in the network. We present here, a low complexity distributed algorithm for detecting and tracking such failures. We assume that nodes inside the failure region are either destroyed or unable to communicate with any other node. The algorithm presented here does not assume global co-ordinate information for the nodes, nor any capabilities to sense the phenomenon causing the failure, and that there are some nodes failing randomly. We utilize the well studied Restricted Delaunay Triangulation (RDT) and show that we can distinguish between systematic and random failures, thus enabling their robust detection. By formulating simple, local optimization problems which have closed form solutions, the evolution of the failure front is tracked accurately. The methodology presented is evaluated using several substantiating simulations.

Topological Analysis In contrast to deployment related failure problems, the research this topic focuses on detection of dynamic failure regions, and analysis of their topological changes. While the related work presented here is in no way comprehensive, we intend to provide an insight into the distributed algorithms for detecting topological changes.

The work in [91, 93], seeks to detect topological changes on a scalar field defined on the nodes in the network. The topological changes are described as creation/loss of holes and formation/merging of connected components. In [91], such events are detected by observing the 1-hop neighborhoods of nodes, and by detecting the topological changes therein, with the assumption that the communication graph accurately captures the topology of the field of interest. [93] constructs a tree structure representing the adjacency of topological components, and by analyzing the change in this structure when the topology changes, are able to detect it.

In [92, 96], the authors seek to track an evolving topological component, for example, a subset of the region where a defined field is above a threshold. The topological component of interest is assumed to be of polygonal shape, which is identified by computing the convex hull of the component. Such computation is facilitated by maintaining a hierarchical structure in the network. The cluster heads gather the required information from nodes in the cluster to compute the convex hull. [96] extends such analysis to cases where the sensor measurements are noisy and clear transition is not well defined. A regression-based spatial estimation technique determines discrete points on the boundary and estimates a confidence band around the entire boundary, and a Kalman Filter-based temporal estimation technique tracks changes in the boundary and aperiodically updates the spatial estimate. They assume global coordinate information in order to perform the regression analysis.
In contrast to the above mentioned articles, our work deals with catastrophic situations with the following characteristics:

1. the nodes inside the region of interest fail completely, or are unable to communicate with any other node,
2. such nodes cannot participate in response mechanisms, restricting the tasks performed to the information available in the neighboring (still active) nodes,
3. lack of hardware with capabilities to sense the phenomenon causing the failure,
4. and, there are "randomly" occurring failures of nodes.

Network model We consider a region $\mathbf{R} \subseteq \mathbb{R}^2$, in which a set of nodes V are deployed. For a communication radius r_c, a communication graph $G = (V, E)$ is induced on V where $(v_1, v_2) \in E \iff d_{12} = |\,(v_1, v_2)\,| < r_c$. An edge (v_1, v_2) in G implies that v_1 is within the communication region (and vice-versa) of v_2, or in other words, can be *observed* by v_2. For a node $v_i \in G$, (or equivalently, $v_i \in V$), we denote by N_i, the neighbors of v_i in G. Given this model for the communication graph, the notion of the deployment region $\mathbf{R}$ may be made precise as follows.

Definition 15. *The* projection *$\hat{C}'$ of a subgraph $\hat{C} \subseteq G$, is a collection of line segments $\overline{v_i v_j}$ in $\mathbf{R}$ such that $(v_i, v_j) \in \hat{C}$.*

Definition 16. *The* outer boundary $\partial(\mathbf{R})$ *is the union of line segments in the projection of* G*, which encloses all the nodes. The region of deployment* $\mathbf{R}$ *is the region enclosed by* $\partial(\mathbf{R})$*.*

Throughout, we use v_i to denote the node with index i and also its position in $\mathbf{R}$. We assume that we neither have any localization, i.e., no global co-ordinates for the nodes are known, nor any global orientation information. We however assume the distances between the nodes (the lengths of edges in the communication graph) is known to us. This information can be obtained from several techniques such as Received Signal Strength (RSS) estimation, time difference of arrival (TDoA), etc. [106]

Systematic Failure In the work presented here, we endeavor to track such failures only using the communication graph and its edge lengths. The resolution of the regions we are able to detect is restricted by the node density. As such, we require the region of systematic failure to be "big enough" relative to the communication radius, so as to be detected, and hence tracked. We now make the above description of the systematic failures mathematically precise.

Definition 17. *For a time evolving smooth and simple curve* $C(s,t) : S \times \mathbb{R} \rightarrow \mathbf{R}$*, a* systematic failure *is defined to be occurring if*

1. $\forall t > 0, v_i \in C(s,t) \Rightarrow v_i$ *fails at time t,*

2. $C(s,t)/\partial(\mathbf{R})$ *is closed,*

3. $\exists t_0$*, such that* $\forall t > t_0$*, we can inscribe a circle of radius* $r_c/\sqrt{3}$ *in the region enclosed by* $C(s,t)$ *(when* $C(s,t)$ *is closed), or by* $C(s,t)$ *and* $\partial(\mathbf{R})$ *(when* $C(S,t)$ *is open and intersects the boundary),*

4. *The area enclosed by* $C(s,t)$ *(when* $C(s,t)$ *is closed), or by* $C(s,t)$ *and* $\partial(\mathbf{R})$*, is increasing with time.*

The time evolution of the curve $\mathbf{C}(s,t)$ can be specified by assigning a velocity vector at each point of the curve in a normal direction. We consider the vector only in the normal direction, as any horizontal component will only result in a reparameterization of the curve. Therefore, the boundary of any systematic, time evolving failure region can be described by a time evolving curve given as:

$$\frac{\partial C(s,t)}{\partial t} = -\nu(s,t)\bar{n}, \tag{66}$$

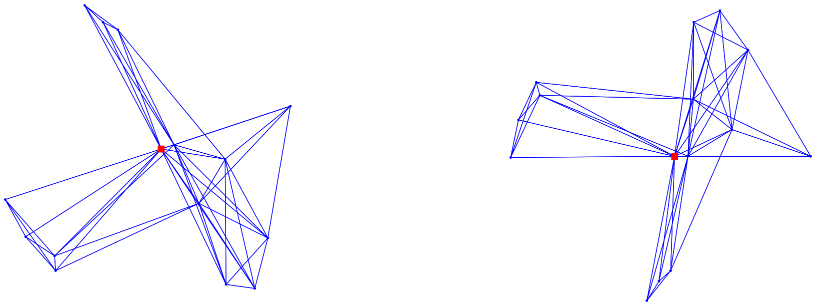
where $\nu(t,s)$ is a scalar specifying the speed with which the curve is expanding at parameter s and time t.
In contrast to a systematic failure, nodes may fail "randomly", in which case, the positions of the failed nodes do not necessarily lie in a region which can be expressed by above properties. In fact, when nodes fail randomly, their positions are usually uncorrelated in space or time.

Algorithm Overview Our process of detecting and tracking the systematic failure has the following steps:

1. compute local coordinates

2. compute restricted delaunay triangulation and its boundary

3. compute tight subgraphs



(a) subgraph induced on G by a node and its neighbors

(b) subgraph as realized according to local coordinates

Figure 46: Comparison of original subgraph (in global coordinates) with the one realized using local coordinates. The subgraph is induced by a node (shown as red block) and its 1-hop neighbors.

4. identify the tight subgraphs which enclose failed nodes (active contours)
5. detect systematic failures and compute the speed of propagation

Local coordinates At each node v_i, given the subgraph $G_i \subseteq G$ induced by the nodes $v_i \cup N_i$, and the distance function D_i indicating the lengths of edges in G_i, we compute of the local coordinates $\mu_i : v_i \cup N_i \rightarrow \mathbb{R}^2$. The local coordinates are computed such that $\|\mu(v_j) - \mu(v_k)\| = D_i(j,k), \forall (v_j, v_k) \in G_i$. Localization in sensor networks, is a well researched subject, with several distributed algorithms presented in the literature [107],[108]. Unlike network localization problems seeking to compute global coordinates for the nodes, we are primarily concerned with coordinates locally at each node and its neighbors. Table 4 describes the algorithm for computing local coordinates and Figure 46 shows an example application of this algorithm.

Restricted Delano Triangulation Restricted Delaunay Triangulation ($RDT(G)$) [109] has the following properties which are useful in our context.

1. $RDT(G)$ is locally similar to DT, in that it is planar, and the circumcircle of a triangle does not contain any other node.
2. Denote by $UDel(G)$, the graph obtained by removing all the edges of length greater then r_c from DT. $RDT(G)$ satisfies the following set inequality:

$$UDel(G) \subseteq RDT(G) \subseteq G$$

A distributed algorithm for computing $RDT(G)$ was given in Chapter 3 of the thesis [110]. An important aspect of this algorithm in the context of distributed processing is that, under dynamic conditions, when a set of nodes fail, the $RDT(G)$ only needs to be recomputed locally. It is also shown that the communication complexity (total number of messages broadcasted over all the nodes) is of the order $O(\sqrt{n \log n})$. Given a triangulation on a manifold, the boundary of the manifold may be found by looking at the edges in the triangulation, which are

Table 4: Algorithm for computing local coordinates μ_i at v_i

```
set $\mu_i(v_i) = (0, 0)$
$if$ there exists a 4-clique including $v_i$
    find the most robust 4-clique
$else\ if$ there exists a 3-clique including $v_i$
    select a 3-clique
$else$
    distribute the nodes evenly around $v_i$
    return
compute coordinates for the nodes in the clique selected
put the collected clique nodes into *poolNodes*

$while$ nodes remaining with no coordinates {
    $if$ there exists a node with 3 links into the *poolNodes*
        triangulate its position
    $else\ if$ there exists a node with 2 links into the *poolNodes*
        choose the node with smallest distance from node $v_i$
        fix the position which conforms with the local adjacency
    $else$
        find a position which conforms with the local adjacency

    put the nodes with computed coordinates into *poolNodes*
}
```

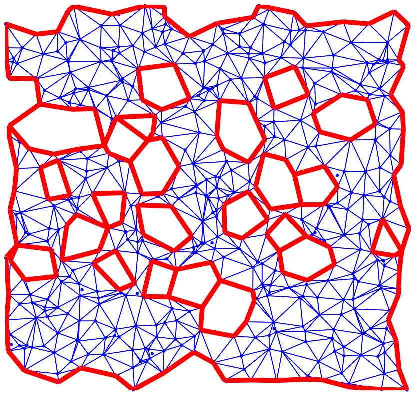


Figure 47: The boundary of the restricted delaunay triangulation of G.

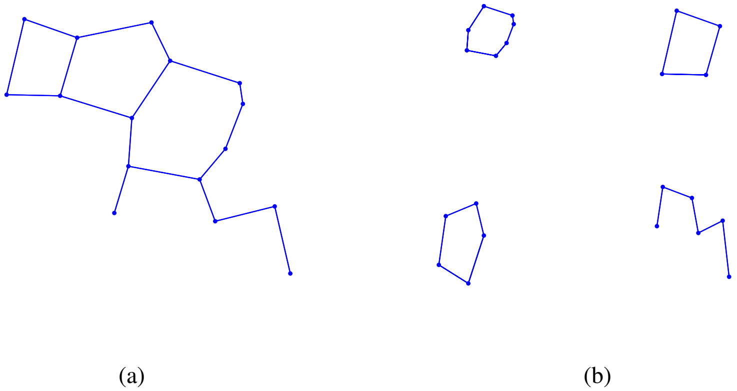


Figure 48: dividing a connected component in $\partial(RDT(G))$ into tight subgraphs.

faces of only 1 triangle. The boundary of $RDT(G)$ gives a close approximation to the boundary of the network, a part of which may potentially describe the propagating failure front. Figure 47 shows $RDT(G)$ and the boundary (in red).

Tight subgraphs. To "tightly" surround a region which might potentially be the systematic failure region, we need to further partition $\partial(RDT(G))$. Given a subgraph with loops and handles as shown in Figure 48(a), our goal is to divide this subgraph into smaller loops and handles as shown in Figure 48(b). ote the outermost boundary of $\partial(RDT(G))$ encloses all the nodes and does not give us any information about the "holes". Therefore, we remove the outermost boundary. The handles in the figure may be caused if the failure region is near the boundary. In all subsequent discussions, we denote $\partial(RDT(G))$ to be the boundary with the outermost boundary removed. The Algorithm given in Table 5 computes tight subgraphs.

Active Contours. We need a test which identifies whether a given tight subgraph is an active contour (i.e., it encloses failed nodes). The main idea is to test whether the failed nodes being observed by the nodes on a tight

Table 5: Algorithm for finding *tight* subgraphs in $\partial(RDT(G))$

```
start
    select a node $v_i$ with degree 2
    send a forward packet $(i, i, 1)$ to one of its neighbors
when a node $v_k \neq v_i$ receives a forward packet $(i, j, h_i(j))$ from node $v_j$
    store in table $(i, j, h_i(j))$
    broadcast to all nodes in $N_k \setminus j$, the packet $(i, k, h_i(j) + 1)$
when a node $v_k \neq v_i$ receives a reverse packet $(i)$
    assign $v_k \rightarrow comp(i)$
    find $\hat{j} = \arg\min_j h_i(j)$
    transmit to $\hat{j}$, the reverse packet $(i)$
at node $v_i$
    wait to receive all the forward packets $(i, j, h_i(j)$
    $if$ received any forward packets
        find $\hat{j} = \arg\min_j h_i(j)$
        send the reverse packet $(i)$ to $\hat{j}$
    $else$
        find the shortest path $p$ between the leaves
        $\forall v_k \in p$, assign $v_k \rightarrow comp(i)$
```

subgraph are located in the region separated by the subgraph. When a failed node does not share such a region with any active node, we say that the failed node lies in the region being separated by the subgraph. The theorem below summarizes this idea.

Definition 18. *We define a total cyclic ordering $\diamond$ on the neighbors N_i of v_i, as the order in which the nodes appear either in clockwise, or in counterclockwise direction.*

***Theorem* 19.** *Let v_f denote a failed node. A subgraph $\hat{C} \subseteq RDT(G)$ which has the following properties*

1. *$\exists v_i \in \hat{C}$, with $deg_{\hat{C}}(v_i) = 2$, such that $v_f \in N_i^{RDT}$,*

2. *$\forall v_i \in \hat{C}, \exists$ an active node $v_{aj} \in N_i^{RDT}$ where $v_{aj} \notin \hat{C}$,*

3. *$\hat{C}$ is a modified tight subgraph*

is an active contour enclosing v_f if and only if,

1. *The edges in $\hat{C}$ are faces of at most 1 triangle*

2. *$\forall v_i \in \hat{C}$ with $deg_{\hat{C}}(v_i) = 2$, there does not exist an ordering of the form $v_j \diamond v_f \diamond v_{aj} \diamond v_k \diamond \cdots$, where $v_j, v_k \in \hat{C} \cap N_i^{RDT}$ and $v_{aj} \in N_i^{RDT}$.*

Velocity Estimation. To estimate the normal direction $\bar{n}(s,t)$ and the speed $\nu(s,t)$, locally, we approximate the curve as a straight line. Specifically, for an observing node v_i, let the normal to the curve at s_0 pass through v_i. In the vicinity of v_i, we approximate $C(s,t)$ as its tangent line at s_0. The problem can then be restated as an estimation of the direction θ which the normal line makes at s_0 with the $x-$ axis in the local coordinate system μ, and the speed $\nu(s_0)$. Denote the unit vector in the direction of the normal line at s_0 (direction of arrival) as $\bar{n}_i = (a,b)$. The computations for estimating the velocity are performed at one of the surviving neighbors of v_i, when the node v_i fails. For each node v_j in the set of neighboring failed nodes $N_i^f \subseteq N_i$, denote the local failed time as $t_j = \tau_j^f - \tau_i^f$. As we do not assume global time synchronization, only the local failure times t_j are available to us. The relationship between the positions of the failed nodes, the direction of arrival, and the speed, can then be given locally as:

$$\bar{n}_i \cdot v_j + \nu t_j = 0 \Rightarrow a x_j + b y_j + v t_j = 0 \tag{67}$$

In the case of a systematic failure, Equation (67) will be satisfied for all failed nodes exactly, and the normal direction with the speed can be computed by solving a system of three linear equations. In a practical scenario, we may have random failures of some of the nodes and errors in the recorded failure times. Therefore, we formulate the estimation of the velocity as the following optimization:

$$(\hat{\bar{n}}_i, \hat{\nu}) = (\hat{a}, \hat{b}, \hat{\nu}) = \arg\min_{a,b,\nu} f(a,b,\nu) = \arg\min_{a,b,\nu} \sum_{j \in N_i^f} (a x_j + b y_j + v t_j)^2 \tag{68}$$

with the following constraint

$$h = a^2 + b^2 - 1 = 0. \tag{69}$$

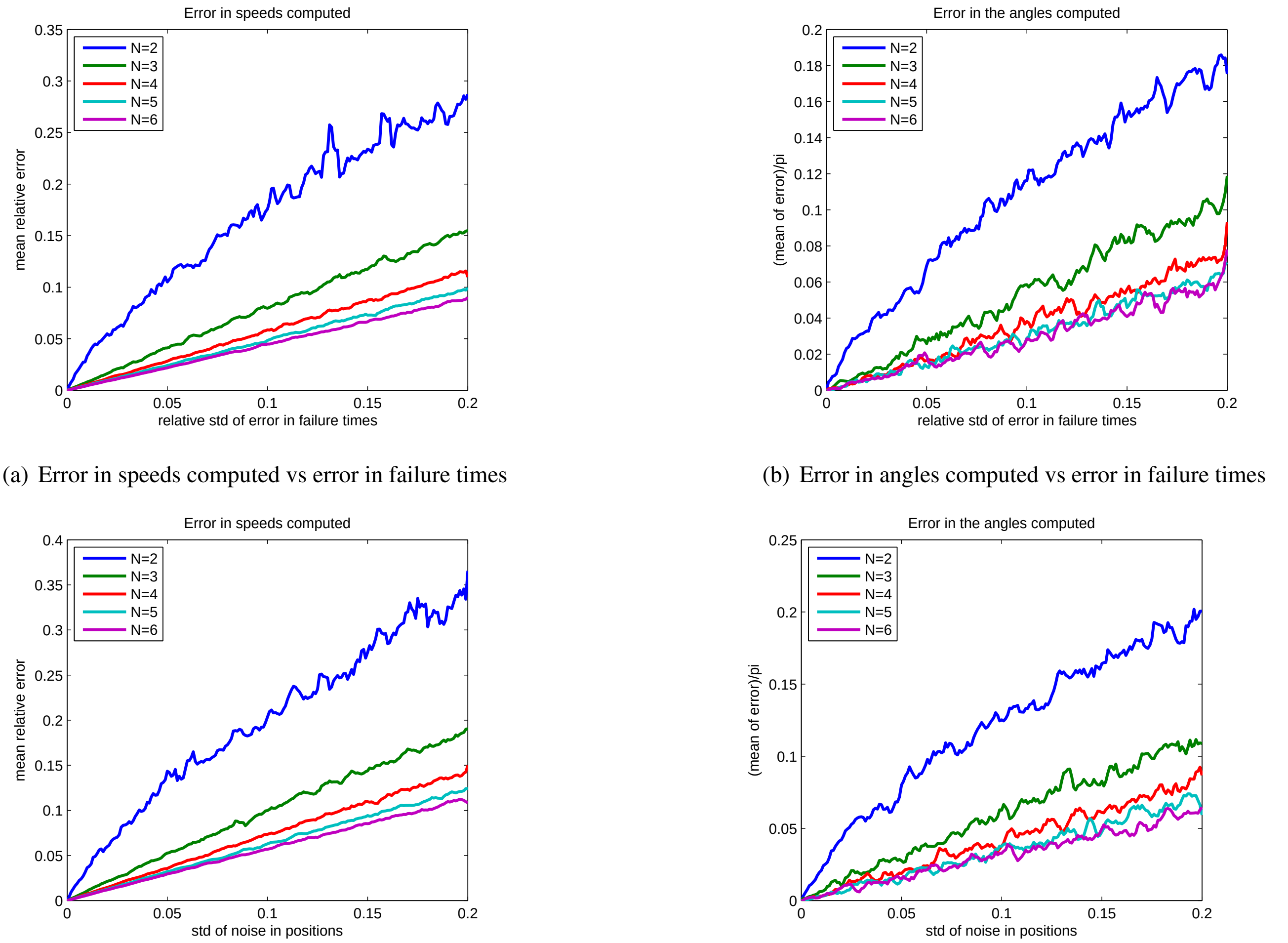
The above optimization problem has a closed form solution. Figure 49 shows some simulations demonstrating the accuracy and robustness of the above method.

5.4.3 Main Results on Network Classification and Failure Detection

In this part, we highlight our experimental results on network classification and synthesis as well as detection results of systematic failures.

Network synthesis. First we present the experimental results on network synthesis. In the first synthesis experiment for our new algorithm, the total time step is 3000 and the initial number of nodes is 4. As shown in Figure 50, the experimental results demonstrate that the corresponding degree distribution of such a model still satisfies a power-law distribution, indicating that the degree distribution is a feature that all resulting networks do share in our new model, as in Barabási-Albert model.

We also record the changes of the statistics of clustering coefficients from time-step 1500, when the changes of statistics of clustering coefficients starting to be very small, i.e., when the statistics has stabilized in the network. Another experiment with the same setting but for Barabási-Albert model is run, and we record the



(a) Error in speeds computed vs error in failure times

(b) Error in angles computed vs error in failure times

(c) Error in speeds computed vs error in node positions

(d) Error in angles computed vs error in node positions

Figure 49: Robustness of the velocity estimation. The error in velocity estimated is small, when the errors in failure times and positions are small, and has a graceful degradation.

changes of statistics of clustering coefficients as well. We calculate and compare the standard deviation of the changes for both experiments, which are normalized by the values of the statistics at the last time-step. The result is shown in Table 6. The changes of the statistics of clustering coefficients in the network generated by the algorithm based on the generalized Markov Graph model are much smaller than those by Barabási-Albert model, which indicates that our new model could generate a network with a much faster stabilizing clustering coefficient distribution than Barabási-Albert model does.

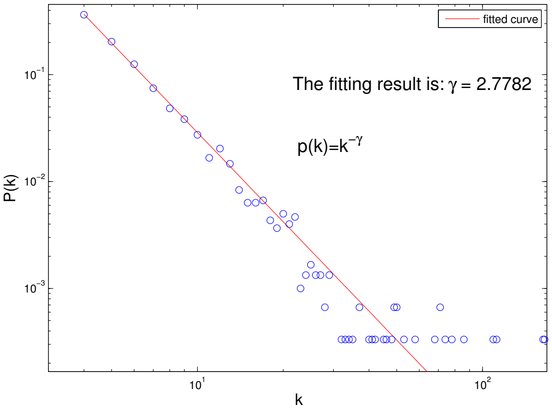


Figure 50: Degree distribution of the network evolving according to the rules that satisfy the properties of the generalized Markov Graph model.

To further validate that the algorithm based on the generalized Markov Graph model outperforms Barabási-Albert model on generating networks with more stable clustering coefficient distributions, we repeat the previous experiment 1000 times for both Barabási-Albert model and our new algorithm. Table 7 shows the mean and variance of the selected statistics (mean, variance, skewness and kurtosis) of the clustering coefficient distributions for both models. Compared to their ranges $[0, 1]$ (as any clustering coefficient is limited between 0 and 1), the variance of the mean and the variance of clustering coefficients for the new model is very small. This verifies that the clustering coefficient distribution is a common feature of all networks generated by our new model. Furthermore, in Table 7 the ratio of the variance to the mean of almost all the statistics of the clustering coefficient distribution for the new model is much smaller than that for Barabási-Albert model. This indicates that the new model could generate networks that have more stable clustering coefficient distribution than Barabási-Albert model does.

In summary, the algorithm based on the generalized Markov Graph model can generate networks that follow a certain degree distribution together with a more stable and faster stabilizing clustering coefficient distribution than Barabási-Albert model does. This is not very surprising in light of the generalized Markov Graph model. The algorithm itself relies on two features of the generalized Markov Graph model, namely the degree distribution and the clustering coefficient distribution. To simulate a network with N nodes, the complexity of this algorithm

models	Barabási-Albert model	generalized Markov Graph model
mean	0.0022	1.97e-4
var	0.0057	7.43e-4
skewness	0.0069	0.0038
kurtosis	0.0221	0.0012

Table 6: Standard deviation of the change of the statistics of clustering coefficients

Algorithm	Preferential Attachment Algorithm from Barabási-Albert model			Algorithm based on generalized Markov Graph model		
Statistics of distribution	mean	var	var/mean	mean	var	var/mean
mean	0.0107	$8.96e-4$	8.37%	$.2279$	$.0011$	0.483%
var	0.0013	$1.507e-4$	11.59%	0.0263	$3.089e-6$	0.012%
skewness	4.83	0.386	7.99%	0.6340	0.09	14.19%
kurtosis	33.8	8.12	24%	3.2171	0.3504	10.89%

Table 7: Statistics of clustering coefficients of networks generated by both models.

is $O(N^4)$, which is the same as Barabási-Albert model. The computational cost of this algorithm is twice that of Barabási-Albert model: both the degree list of the whole network as well as the number of triads at each node have to be calculated in the new algorithm while, in Barabási-Albert model, only the degree list needs to be calculated.

Detection of systematic failures. An active contour is a potential candidate for representing a systematic failure, as it represents spatially correlated failures. However, it might also be surrounding randomly failed nodes. In order to accurately detect a "systematic" failure, we also need to test for temporal correlation, which can be summarized in the following two properties

1. *Causality* in failure times

2. *Consistency* in speeds

By causality we mean, that when seen in the direction of arrival of the propagating failure, the nodes farther away (or are well inside a failure region) should have failed before the nodes which are closer to the boundary. The optimization problem described earlier, estimates both speed $\hat{\nu}_i$ and direction $\hat{\hat{n}}_i$ of propagation at a failed node v_i. The directional distance of a failed node v_j, in the direction of the propagating failure may be determined as the dot product $< v_j \cdot \hat{\hat{n}}_i >$, using the local coordinates at node v_i. The problem then reduces to checking if the order of failure times is the same as that of the directional distances. If the order is not the same, the causality condition is not satisfied, and we consider the failed node v_i as a randomly failed node. If all the failed nodes being observed by the active contour fail the causality test, then the active contour is considered non-representative of a systematic failure.

We also assume that in a systematic failure, the speed of propagation does not change suddenly in any given direction, a property we call *consistency* in speeds. We test for consistency in speeds as follows. Consider a node v_i which fails at time t and is enclosed by an active contour, and v_i estimates the speed of propagation as $\hat{\nu_i}$ (note that this computation is performed at an active node). Let v_{ac} be the node on the active contour which is closest to v_i. Then, if the node v_{ac} is still active at time $t + r_c/\nu_i$, we declare that the failure of node v_i is not systematic. If all the failed nodes being observed by the active contour fail the consistency test, then the active contour is not considered to be representing a systematic failure.

5.4.4 Tracking Systematic Failures

Sensor networks are ideal when deployed to monitor hazardous environments such as volcanic eruptions, wild fires, land slides, war zones etc. A common feature among these environments is that they produce events which cause *correlated failures in both space and time*, which we call *systematic*. Being able to detect and track such failures becomes crucial both for the sake of tracking itself and for any emergency response thereafter. The problem of tracking systematic failures is exacerbated when some of the nodes fail randomly (due to some malfunction), which is the case considered here.

As adopted in this paper, a natural methodology for tracking such systematic phenomenon is based on fast and efficient identification of boundaries. We would expect any algorithm performing this task to include the following important properties: 1) the boundary output is geometrically close to the actual boundary, and 2) the interior of the boundary is topologically faithful to the original space. It is often the case that we are only given random samples from the space. We may then reconstruct the space by first placing balls of a certain radius around these points, and then by taking the union of these balls.

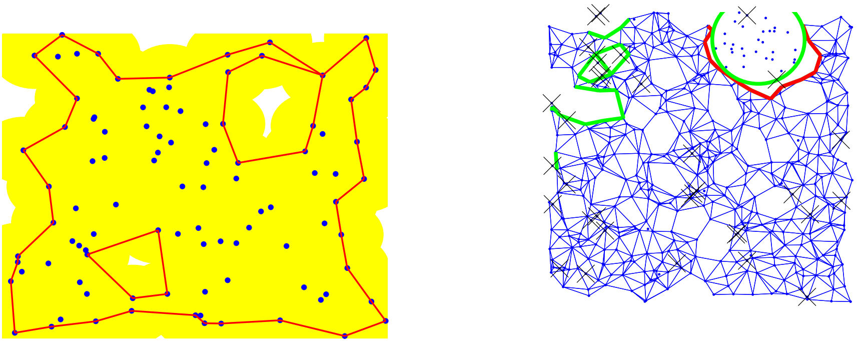

(a) (b)

Figure 51: The systematic failure tracking in done in two steps, (1) updating the boundary of the network at a specific time as shown in (a), and (2) segmenting the boundary and identifying the parts surrounding a systematic failure.

Problem formulation We consider a compact region $\mathbf{R} \subseteq \mathbb{R}^2$, in which a set of nodes V are deployed. For a communication radius r_c, a communication graph $G = (V, E)$ is induced on V where $(v_1, v_2) \in E \iff d_{12} =$

$|(v_1, v_2)| < r_c$. An edge (v_1, v_2) in G implies that v_1 is within the communication region (and vice-versa) of v_2, or in other words, can be *observed* by v_2. For a node $v_i \in G$, (or equivalently, $v_i \in V$), we denote by $\mathcal{N}_i$, the neighbors of v_i in G. Given this model for the communication graph, the notion of the deployment region $\mathbf{R}$ may be made precise as follows.

Definition 19. *The* projection $\hat{C}'$ *of a subgraph* $\hat{C} \subseteq G$, *is a collection of line segments* $\overline{v_i v_j}$ *in* $\mathbf{R}$ *such that* $(v_i, v_j) \in \hat{C}$.

Definition 20. *The* outer boundary $\partial(\mathbf{R})$ *is the union of the minimum number of line segments in the projection of G, which encloses all the nodes. The region of deployment* $\mathbf{R}$ *is the region enclosed by* $\partial(\mathbf{R})$.

We define the *communication region* $\mathcal{R}_c$ as the union of balls of radius $r_c/2$, with centers as the nodes in V. Note that when $\mathcal{R}_c$ is path connected, the communication graph is a connected graph.

Throughout, we use v_i to denote the node with index i and also its position in $\mathbf{R}$. We assume that we have neither localization, i.e., no global co-ordinates for the nodes are known, nor global orientation information. We, however, assume the distances between the nodes (the lengths of edges in the communication graph) are known.

We next make the above description of the systematic failures mathematically precise.

Definition 21. *For a time evolving smooth and simple curve* $C(s,t) : S \times \mathbb{R} \to \mathbf{R}$, *a* systematic failure *is defined if*

1. $\forall t > 0, v_i \in C(s,t) \Rightarrow v_i$ *fails at time t,*
2. $C(s,t)/\partial(\mathbf{R})$ *is closed,*
3. *the region enclosed by* $C(s,t)$ *(when* $C(s,t)$ *is closed), or by* $C(s,t)$ *and* $\partial(\mathbf{R})$ *(when* $C(s,t)$ *is open and intersects the boundary), is large enough to create a hole in* $\mathcal{R}_c$.
4. *The area enclosed by* $C(s,t)$ *(when* $C(s,t)$ *is closed), or by* $C(s,t)$ *and* $\partial(\mathbf{R})$, *is increasing with time.*

The time evolution of the curve $\mathbf{C}(s,t)$ can be specified by assigning a velocity vector at each point of the curve in a normal direction. We only consider the vector in the normal direction, as any horizontal component will only result in a reparameterization of the curve. The boundary of any systematic, time evolving failure region can therefore be described by a time evolving curve given, that is given as:

$$\frac{\partial C(s,t)}{\partial t} = -\nu(s,t)\bar{n}, \tag{70}$$

where $\nu(t,s)$ is a scalar specifying the speed the curve is expanding at, for parameter s and time t, and $\bar{n}$ is the normal vector.

As shown in Figure 51(b), we correctly identify the segment of the boundary surrounding the failure in the presence of random failures, using just the local orientation information. The speed of evolution can then be estimated using an optimization function which has a closed form solution, and which is quite robust to noise as shown in Figure 49. Parts of this work appeared in [111], and in [112], and the complete version of the journal paper is currently under review. A draft is available online at [113].

Summary. The tracking of systematic failure is divided into two steps, 1) Fast and localized estimation of network boundary at each time step, and 2) Robust estimation of segment of the boundary surrounding the failure in the presence of noise, and estimating the speed of evolution. We use a geometric object called an α-shape to define the boundary of the network, and argue for why such a definition is relevant in the context of failures. In addition, we also develop a fast, localized and coordinate free algorithm for computing the α-shape, which has interesting implications on its own. Figure 51(a) shows the α-shape of a network. Observe that it is topologically faithful to the boundary of the communication region. As a by-product of our analysis, we also develop a mathematical framework which unifies several boundary definitions adopted in the literature.

5.4.5 Relevance to Original Goals

To facilitate classification of networks and identify anomalous networks, we introduced a generalized Markov Graph model and its application to social network classification and social network synthesis. The main result of the generalized Markov Graph model is that the degree distribution, the clustering coefficient distribution, and the crowding coefficients are three fundamental statistics for characterizing generic social networks. In addition, the generalized Markov Graph model provides a new insight into clustering coefficient: it is the result of the dependence between higher order structures, namely the triads, in social networks.

To detect and track systematic failures in networks, We have presented a distributed algorithm to track a systematic failure in sensor networks which is robust to random failures. The algorithm presented has a low communication complexity, and requires a small number of local communications which makes it ideal for real time applications. We assume neither any global coordinate information nor any capability of directly sensing the phenomenon causing the failure. We presented precise mathematical formulations of the algorithms along with proofs for their correctness. The simulations performed demonstrate the accuracy and robustness of our methods.

6 Cascading Failures in Power Grids due to Communication and Cyber Attacks

The second issue that our research explores is the formation and properties of correlated failures in communication networks. Following a WMD event, one or multiple failures can be identified. Unfortunately, this is not the end of story: the intrinsic nature of *networking* and *communication* will impulsively surrender a communication network vulnerable to these failures, and further create more and more failures, which is recognized as an *epidemic* and detrimental phenomena in every inter-dependent architecture, including social networks and transportation systems. The majority of existing studies address specific failures as *a-priori* knowledge and design their countermeasures. However, there are very few studies on the evolution process of these failures, and on the extent of damage to network composition and structure. Therefore, this is the first effort in modeling and exploring the impact of correlated failures, which would greatly contribute to the failure resilience of large networks.

By analyzing the correlated failures, we modeled the impact of traffic overloading. To further study the performance impact of failures, we need address the performance of cyber-physical attacks with new metrics. More over, we aime to study the scale properties of attacks. The detection and localization of failures provide the information of where a failure occurs and what its impact is on the topological composition. However, failures in the same location may have varying impact because of their attack methods. We seek answers to questions like 1) what kind of properties should an attack have in order to make a severe damage to the communication networks? 2) Are there any asymptotic bounds of an attack in temporal domain? 3) What is the scaling law in large-scale networks?

6.1 Performance Impact of Cyber Attacks

6.1.1 Objectives and Approaches

The cyber-physical attacks, not only sabotage the vulnerable or impaired nodes, but also reproduce themselves to propagate the damage epidemically. On the other hand, the damage to communication links, or *physical* in contrast to *cyber* attacks, may be easily detected. However, it is a challenging issue to assess the impact of such destruction. However, to quantify the network vulnerability in tempo-spatial domain, the widely used performanc e metrics, such as throughput or delay cannot not illustrate the impact of failures. Therefore, we need to propose new metrics, and more important a new method to study the problem.

Our methodology is to study the gain that a misbehaving node can obtain via two general classes of backoff misbehavior. The first class is called *continuous misbehavior*, which performs misbehavior persistently and does not stop until it is disabled by countermeasures. The second class is called *intermittent misbehavior*, which in contrast to continuous misbehavior, performs misbehavior in *on* periods and returns to be legitimate in *off* periods. The goal of intermittent misbehavior is to obtain benefits over legitimate nodes and at the same time to evade the misbehavior detection. Then, we define both legitimate users and misbehaving users formally in the sense that their attack time and selection of attacking are represented by mathematical models.

One of the main contributions of our work is to define a new metric, namely, “order gain” to quantify the benefits of backoff misbehaving nodes, or the damage to legitimate users. Then based on this measure, we examine the responses of different attack models and identify the most harmful attack models for which the countermeasures need to be designed.

Main Results: Our contributions in the study of A new metric, order gain, is defined to measure the performance benefits of misbehaving nodes over legitimate nodes, which is helpful in evaluating the gain and impact of a misbehaving node in a CSMA/CA-based wireless network. We find that the order gain of a continuous double-window backoff misbehaving node always converges to $\log_2(p/p_D)$ as $t \to \infty$, where p and p_D are the collision probabilities of legitimate and misbehaving nodes, respectively. While the order gain of a continuous fixed-window backoff misbehaving node is an increasing function to infinity as $t \to \infty$. We also find that the order gain of the intermittent misbehaving node always converges as $t \to \infty$ regardless of the misbehavior scheme it chooses in the *on* state.

Our analytical and experimental results show that both double-window backoff misbehavior and fixed-window backoff misbehavior achieve significant gains when the number of users is small. However, double-window backoff misbehavior is more sensitive to the number of users and has marginal gains as the number of users increases. Thus, the number of users can be considered as an evaluating factor for the deployment of a counter-strategy in a

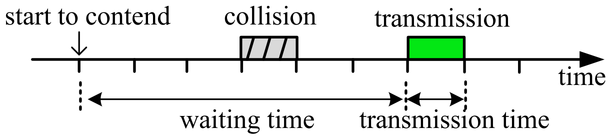


Figure 52: A single transmission in a simple slotted CSMA/CA network

wireless network. We also find that an intermittent misbehaving node can not achieve substantial gain by setting a short *on* period to perform misbehavior. Thus, even an intermittent misbehaving node may evade misbehavior detection, it can not cause significant damage to a wireless network.

6.1.2 What is Order Gain?

The benefits of misbehaving nodes can be either gaining more resources for selfish nodes or degrading network performance. In the first case, a selfish node tries to acquire a higher chance to access the channel than legitimate nodes, which is quite easy in operation. However, the effect of misbehaving can be devastating, and therefore, it is one of the earliest and most well studied issue for wireless access networks [**?**, **?**, **?**, **?**]. In the second case, the goal of malicious nodes is to disrupt normal network operation. Such nodes are often referred to as jammers [**?**, **?**]. In this work, we focus on the first case that in fact would evolution into the second case under condition.

The network performance, on the other hand, can be evaluated by a number of metrics, such as throughput, which can be the data transmission rate of one user, or aggregated rate of a group of users. There have been many works on throughput analysis of CSMA/CA networks, such as [**?**] and [**?**]. By taking a close look, we can find that many analysis are based on the waiting time and transmission time. For example, Figure 52 illustrates a simple example of a transmission in a slotted CSMA/CA network. During the transmission in Figure 52, the throughput can be computed as $\eta = \text{transmission time}/(\text{waiting time} + \text{transmission time}) = 1/7$. We can see that the throughput η is in fact a consequence of waiting time that is the number of slots during the node contends for the channel. Therefore, the waiting time can immediately represent the performance of a node: the longer the waiting time, the worse the performance, and vice versa. We define the waiting time as follows.

Definition 22 (Waiting-time)**.** *Waiting time of a node is the number of slots between the instant that the node starts to contend for the channel to transmit a packet and the instant that the node successfully transmits the packet; namely, the waiting time $W \triangleq \sum_{i=0}^{N} T(i)$, where N is the number of collisions before the node makes a successful transmission, $T(i)$ is the random backoff time after the i-th collision.*

We have shown that the waiting time is essential to the performance of a node. However, our objective is not to evaluate the performance of a single node but to understand benefits of backoff misbehaving schemes, that is the *gains* of misbehaving nodes over legitimate nodes. To achieve this goal, we will introduce a performance metric by considering the following constraints:

- The definition of the metric must be generic, without depending on a particular protocol. This is due to the wide deployment of CSMA/CA networks, especially IEEE 802.11 and IEEE 802.15. Therefore, the definitions of control messages, such as RTS/CTS, ACK should not affect the interpretation of the gain.

- The metric can not be limited to the first-order statistics, which are referred to as the means of random variables, such as mean delay. First-order statistics are in general not able to be extended to other performance metrics. For example, the delay jitter [**?**], which is a second-order statistic, can not be calculated by only knowing the mean delay.

- If the gain of node A over node B is G_1 and the gain of node B over node C is G_2, then the gain of node A over node C is G_1+G_2. This property is very important because it enables us to quantitatively compare the impacts of two misbehaving nodes by directly comparing their metrics.

With above consideration, we introduce a new metric, namely order gain of waiting time[4] as follows.

Definition 23 (Order gain of waiting time)**.** *Let W_A and W_B be the waiting times of nodes A and B, respectively. The order gain of node A over node B is defined as*

$$G(t) \triangleq \log_t \frac{\mathbb{P}(W_B > t)}{\mathbb{P}(W_A > t)}, \tag{71}$$

where $\mathbb{P}(W_A > t)$ and $\mathbb{P}(W_B > t)$ are the tail distribution functions (or complementary cumulative distribution functions, CCDFs) of W_A and W_B, respectively.

Remark 15. *The definition of order gain is based on tail distribution functions of nodes A and B. The tail distribution function, for example, $\mathbb{P}(W_A > t)$ denotes the probability that the waiting time of node A is greater than a given t, showing that how often the waiting time of node A is larger than a given value. Thus, $\mathbb{P}(W_A > t)$ can in fact indicate the performance of node A since the larger the waiting time, the less the chance for the node to access to channel.*

The most commonly-used misbehaving backoff schemes are double-window and fixed-window misbehavior, which both belong to continuous misbehavior and have been extensively studied regarding detection schemes [**?**, **?**] and incentive-based protocols [**?**, **?**]. Therefore, in this section, we first study the two continuous misbehavior: double-window misbehavior, which conforms to binary exponential backoff but chooses a smaller minimum contention window than legitimate nodes, and fixed-window misbehavior, which chooses random backoff time uniformly from a fixed range. Then, we move on to the intermittent backoff misbehavior, in which a misbehaving node performs misbehavior and legitimate backoff in *on* state and *off* state, respectively.

We summarize the main results on the order gain of double-window misbehavior as follows.

***Theorem* 20.** *The order gain of a double-window backoff misbehaving node over legitimate nodes is*

$$G_D(t) = \log_2\left(\frac{p}{p_{_D}}\right) + \Theta\left(\frac{1}{\ln t}\right),^5$$

where p and $p_{_D}$ are the collision probabilities of the legitimate and misbehaving nodes, respectively.

[4] This is referred to as order gain throughout this paper unless otherwise specified.

[5] We say function $f(x)$ is of the same order as function $g(x)$ and write $f(x) = \Theta(g(x))$ if and only if there exist two positive real numbers c_1 and c_2 and a real number x_0 such that $c_1|g(x)| \le |f(x)| \le c_2|g(x)|$ for all $x > x_0$.

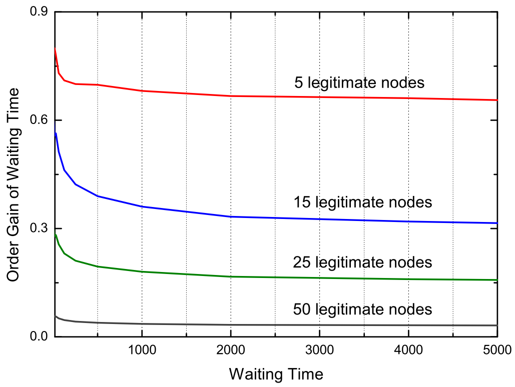


Figure 53: Order gain of a double-window backoff misbehaving node in an 802.11 network with different numbers of legitimate nodes.

Remark 16. *Theorem 20 shows that $G_D(t)$ converges to $\log_2(p/p_D)$ as $t \to \infty$, showing that the order gain of double-window misbehavior depends on the ratio of the collision probabilities of legitimate and misbehaving nodes. It has been shown in [**?**] that the ratio $p/p_D \to 1$ as the number of nodes goes to infinity in a network, which in turn indicates that the performance gain of a double-window misbehaving node becomes marginal as the number of nodes increases.*

Remark 17. *We also find that the order gain of fixed-window backoff misbehavior is an increasing function to infinity as $t \to \infty$ regardless of the number of nodes in the network, which indicates that a misbehaving node can always obtain substantial benefits from fixed-window backoff misbehavior. Thus, any countermeasure to backoff misbehavior should consider fixed-window backoff misbehavior as its primary target.*

6.1.3 Intermittent Backoff Misbehavior

We have derived the order gains of the two widely-used schemes for continuous misbehavior. However, a misbehaving scheme is not always guaranteed to be continuous, especially when there exists a counter-strategy in the network that tries to detect and to disable any misbehavior. It has been shown in [**?**] that a node that performs misbehavior intermittently may evade such misbehavior detection. Thus, it is important to know the gain of an intermittent misbehaving node in a network. The backoff scheme of an intermittent misbehaving node is defined as a Markov process with *on* and *off* states. With this definition, we state our result on intermittent misbehavior.

***Theorem* 21.** *The order gain of an intermittent misbehaving node over legitimate nodes satisfies*

$$G_I(t) = \log_2 \frac{p_{on}}{p_{off}} + \Theta\left(\frac{1}{\ln t}\right),$$

where p_{on} and p_{off} are collision probabilities of legitimate nodes in on *and* off *states, respectively.*

Theorem 21 shows that, perhaps surprisingly, the order gain of an intermittent misbehaving node $G_I(t)$ always converges as $t \to \infty$ regardless of the misbehaving backoff scheme in the *on* state.

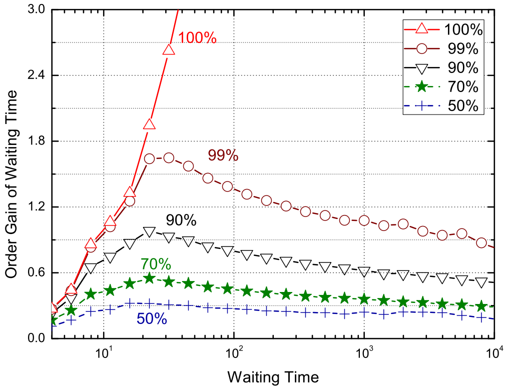


Figure 54: Order gain of an intermittent misbehaving node in an 802.11 network with 5 legitimate nodes.

We use ns2 simulations to further assess the performance of intermittent misbehavior by considering an 802.11 network consisting of five legitimate nodes and one intermittent misbehaving node. The intermittent misbehaving node performs misbehavior by choosing random backoff time uniformly from $[0,7]$ when it is on. Figure 54 demonstrates the order gains of the intermittent misbehaving node for different on-state ratios θ. We see from Figure 54 that the order gain of the misbehaving node always has an initial increasing phase, and after reaches a maximum, starts to converge decreasingly, which shows there exists a phase transition phenomenon in the order gain of intermittent misbehavior. The phase transition phenomenon is more evident when θ becomes large. We denote by t^* the phase transition point, which is the value of waiting time corresponding to the maximum of the order gain. During simulations, we find that t^* increases as θ increases, but the increment is not significant. For example, in Figure 54, t^* increases from 18 to 33 as θ goes from 50% to 99%.

Figure 54 also shows that the order gain of the intermittent misbehaving node is not significant when θ is small. For example, when $\theta = 50\%$, the order gain is always smaller than 0.35 and the phase transition phenomenon is not evident. When $\theta = 70\%$, the order gain is also upper bounded by 0.6. Consequently, our simulation results indicate that if an intermittent misbehaving node tries to evade misbehavior detection by choosing a small θ, its performance gain is not significant.

The goal of an intermittent misbehaving node is to achieve performance gain over legitimate nodes and at the same time to evade misbehavior detection. We find that, interestingly, if an intermittent misbehaving node chooses a small θ to evade misbehavior detection, it can not achieve substantial gains. For example, the order gain is always smaller than 0.35 when $\theta = 50\%$. On the other hand, if an intermittent misbehaving node chooses a large θ to achieve substantial gains, it may not be able to evade misbehavior detection in that it performs similarly as a continuous misbehaving node. For example, we can see that in Figure 54 that when the intermittent misbehaving node has $\theta = 99\%$, it has the similar order gain as the "100%" order gain when the waiting time t is small, which shows that it has a higher risk to be detected.

6.1.4 Discussions and Summary

So far, we have studied the problem of quantifying the gain of backoff misbehavior and obtained the order gains for two continuous backoff misbehavior schemes and the intermittent misbehaving scheme, which are validated by simulation. We further present experimental results to illustrate the impact of backoff misbehavior. Our findings can be summarized as:

1. Double-window misbehavior is more sensitive to the number of users than fixed-window misbehavior and can only achieve marginal gains when the number of user increases, which shows that, on the other hand, the performance loss of legitimate nodes due to double-window misbehavior is not significant in a network with a large number of users.

2. Fixed-window misbehavior can always achieve substantial gains over legitimate nodes regardless of the number of users. Therefore, fixed-window misbehavior should always be the primary target of countermeasures to backoff misbehavior.

3. An intermittent misbehaving node can not achieve significant gain when it chooses a small θ to evade misbehavior detection.

The above results are studied from a "gain" perspective. Note that the network resources are limited and finite, especially for a number of users sharing the same medium. In other words, when some users gain throughput or bandwidth benefits, others can potentially lose their transmission opportunity, resulting in zero user-throughput. A trivial example is that one user occupies the channel for the entire time period, regardless of transmitting useful data or not, which is an extreme of misbehavior and becomes so called "jamming" attack [**?**]. When this happens, the entire network appears to be dysfunctional, and even not accessible to legitimate nodes. It is interesting to see that the distribution function of a jammer's waiting time $\mathbb{P}(W_J > t) = 0$ for all $t > 1$ since the jammer never backs off. Then, a jammer's order gain $G_J(t) = \infty$ for all $t > 1$, showing that the jammer has "infinite gains" over legitimate nodes.

It is worthy of mention that our results have several limitations: 1) We did not consider the upper limits of contention window and re-transmissions for legitimate nodes, such as the 7 short-retry limit in the basic access model of 802.11 DCF. Thus, the order gain is in fact a theoretical metric to performance gain of backoff misbehaving nodes. Nevertheless, we believe our results are still applicable in practical scenarios. For instance, a legitimate node will start a new transmission after reaches the upper limit of re-transmissions, which means its chance to access the channel becomes larger. Thus, our results should provide an upper bound on performance gain of misbehaving nodes for a practical network. 2) Our experiments are limited in a small-scale and single-hop network. Thus, our experimental results may not be able to reveal the performance and impact of misbehaving nodes in more complicated wireless environments. 3) We acknowledge that in the real world, a misbehaving node may be more sophisticated than our models defined in this paper. However, our analytical framework can still serve as a preliminary study to more complicated misbehavior models.

Cyber-attacks, such as virus, not only sabotage the vulnerable or impaired nodes, but also reproduce themselves to propagate the damage epidemically. On the other hand, the damage to communication links, or *physical* in contrast to *cyber* attacks, may be easily detected. However, it is a challenging issue to assess the impact of such destruction.

This study provides another perspective to understand the scale properties of attacks. The detection and localization of failures provide the information of where a failure occurs and what its impact is on the topological composition. However, failures in the same location may have varying impact because of their attack methods. We seek answers to questions like 1) what kind of properties should an attack have in order to make a severe damage to the communication networks? 2) Are there any asymptotic bounds of an attack in temporal domain? 3) What is the scaling law in large-scale networks?

6.2 Tracking Power Grid Vulnerability under Data-Centric Attacks

Smart grid is a cyber-physical system, which integrates communication networks into traditional power grid. This integration, however, makes the power grid susceptible to cyber attacks. One of the most distinguished challenges in studying the aftermath of cyber attacks in smart grid is referred to as *data-centric* threats. In power grids, these data-centric attacks may result in unstable power systems, and further detrimental impact of power supplies. In this paper, we present *Greenbench*, a benchmark that is designed to evaluate real-time power grid dynamics in response to data-centric attacks. The simulation results provide several counter-intuitive suggestions to both smart grid security research and deployment

6.2.1 Motivation

As a prospective replacement to the traditional power grid, smart grid promises a more reliable, effective and efficient power delivery and distribution by integrating advanced communication technologies into traditional power grid. This integration, however, brings a new host of vulnerabilities stem from Internet and opens the door for potential adversaries to tear down a physical system through a cyber attack.

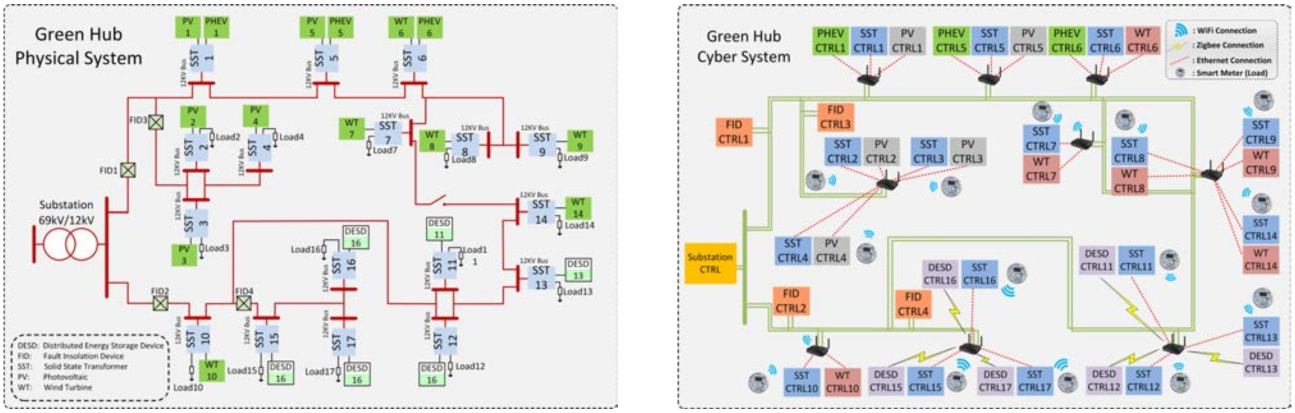
Being aware of the risks, researchers begin to study potential cyber attacks and develop defense schemes to protect this cyber-physical system [114, 115]. However, a practical security solution remains daunting partly because the lack of a commonly recognized platform to evaluate the attack/defense scheme. Question arises when we try to classify various attacks so that we could develop protection solutions in a prioritized way: *How do we analyze, simulate, and evaluate the physical impact caused by a cyber attack in smart grid?*

To address this question, we focus on the *data-centric* threats in smart grids. A data-centric attack in cyber system aims at gaining advantage by manipulating data exchanged within this system. Although vary in form, the basic attributes of data-centric attacks always lies in one or more of the three categories: Confidentiality, in which the attacker gains access to data which is not supposed to be disclosed to him; Integrity, in which the attacker *distort* the content of data; and Availability, in which the attacker *block* or *delays* the data delivery to legitimate user. These three attributes are the basis of information security and the breach on any of them may cause disastrous consequence.

Even though such attacks are critical to the information network, they will result in much more cascading impact than they behave in cyber world. This is because for an information-centric network, distorted or delayed information undermines services and applications. But in power grid, these data-centric attacks may result in bursty traffic of power flows, unstable power systems, and further detrimental impact of power supplies.

Critical as they are in the cyber domain, the impact and destructiveness of date-centric attacks could be amplified significantly when being brought into cyber-physical systems like smart grid. From academic researches such as the false data injection attack [116] which points out the design flaw of the monitoring system in modern power grid, to practical attacks like the Stuxnet [117] which destroys nuclear power plant by infecting and distorting control data, it is obvious that data-centric cyber attacks is real and the demand for the defense is urgent.

6.2.2 Green Hub: A Micro Smart Grid



(a) Green Hub Physical System

(b) Green Hub Cyber System

Figure 55: Cyber-physical system.

Our objective is to develop a *cross-domain* simulation platform that can be used to demonstrate the interaction and inter-dependency of cyber attacks and power grid in real-time. As a platform, we use *Green Hub* as the underlying physical system for our study.

The Green Hub system is a novel distribution level microgrid which has been developed at the Future Renewable Electric Energy Delivery and Management (FREEDM) systems center for the study of power management strategies [118]. The Green Hub is abstracted from an actual residential distribution system, which is a 230kV/22.86kV substation along with two 22.86kV distribution feeders in the Raleigh area, while the substation voltage is reduced to 69kV/12kV to fit our study purpose. The Green Hub contains various innovative power devices developed in FREEDM center, such as the Solid State Transformer (SST), and the Fault Isolation Devices (FIDs), and it is also connected to green energy sources such as the Photovoltaic (PV) and Wind Turbine (WT). All those devices are equipped with Intelligent Electronic Devices (IEDs), which are ARM-based embedded systems used for real time control/monitor and communication. Those IEDs interact with each other to make the Green Hub a self-autonomous micro smart grid which could either be connected to main power grid or operate in an isolated mode.

In order to use this power subsystem for our study, we have to deal with two issues as follows:

- **System abstraction:** An actual system includes a large number of various devices which makes it improper for study and simulation, thus it is necessary to simplify and abstract a high-level system with a suitable size and omit the minor details. The abstracted power system is shown in Figure 55(a), which is a 17-bus power distribution system. Each bus is connected with a SST, which is able to implement bi-directional energy flow and DC/AC transformation. Each SST is connected with a load (Load represents AC load, PHEV represents DC load), and a renewable energy source (PV, WT, or DESD). To ensure the reliability of the system, four FIDs are deployed on different feeder segments, which will open the circuit breaker and isolate failure from upper level power grid in case of a fault happens.

- **Domain mapping:** The challenge here is to map the physical domain into cyber domain by replacing each physical devices with its corresponding IEDs, the mapped cyber domain system is shown in Figure 55(b). Smart meter is used to represent AC load as it is the typical controller for AC load such as households or buildings. Also shown in this figure is the different network access methods for various IEDs (controllers), which reflect the enabling works undergoing in FREEDM center. Specifically, the SST, PHEV, PV and WT controllers are connected to the communication network using Ethernet, the DESD controller is connected using Zigbee, and the smart meter uses wireless to access the network.

The framework of *Greenbench* with its software implementation architecture is shown in Figure 56. We briefly introduce the architecture and the functionality of each block of the framework, while leave detailed description and design challenges to next section.

The *Greenbench* framework is functionally composed by two parts (simulators), the physical part (PSCAD) and the cyber part (OMNeT++). The physical and cyber domain model shown in Figure 55 is built in their corresponding part, and the two parts interact through two interfaces, the *interactor*, and the *buffer files*.

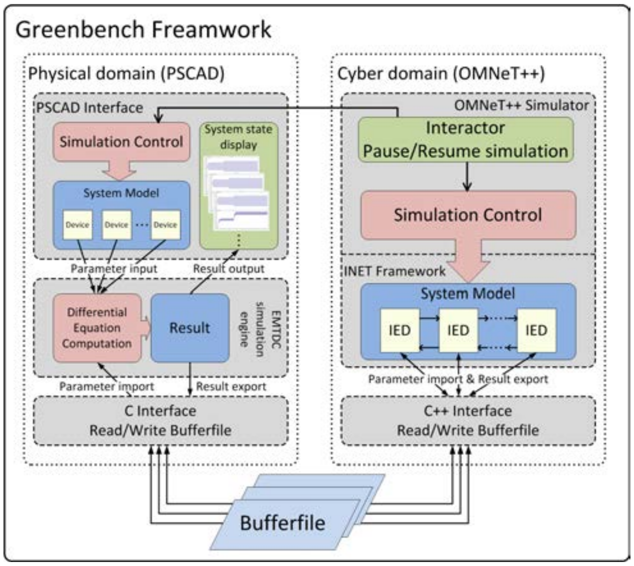


Figure 56: Software implementation of *Greenbench*.

6.2.3 Delayed and Distorted Data-Centric Attacks

In the data-centric attack, the attacker aims at gaining advantage or cause damage by manipulate the data exchanged between network entities. This data-centric attack is even more dangerous in smart grid because instead of interrupt applications and services in cyber world, it will disturb and damage the critical infrastructure, and potentially cause disastrous loss which is not confined only in terms of economic. In order to better understand its impact, find effective solutions as well as instructive suggestions, we hereby study the data-centric attack in smart grid by focusing on smart meter targeted attacks.

Smart meter in AMI is one of the most vulnerable components in smart grid. For the first reason, it is physically accessible to public; For the second , it uses wireless communication which is susceptible to jamming attack [119] and easy to be overheard [120]; For the last and most important, it is usually overlooked by manufactures and was not designed to resist any cyber attack [121, 122]. However, a system is as strong as its weakest link, and it remains an open question that *whether the omitted security feature on smart meter is reasonable*. To address this question, we select three cases from different security aspect and study them in *Greenbench*, which include a delayed data attack, a distorted data attack, and a composite attack.

The metrics usually used to observe the state of a power system is voltage, current, real power and reactive power. For the simulated power system, the voltage on each point will remain unchanged unless an overload happens, nut the current keeps changing with variation of load; while the trend for real power and reactive power change follow the same pattern during our simulation. Therefore, we use current and real power to illustrate the state change of the Green Hub hereafter.

For easy description, we divide the Green Hub shown in Figure 55(a) into 4 sections: Section 1 starts after FID1 and includes load 1, 5, 6, 7, 8, and 9; Section 2 starts after load 10 and includes load 11, 12, 13, and 14; Section 3 starts after FID3 and includes load 2, 3, and 4; And section 4 starts after FID4 and includes load 15, 16, and 17. Note that load 10 does not belong to either sections.

Delayed Price Information in AMI In this case study we simulate and analyze the "jamming the price signal attack" which was proposed in [119]. Particularly, it is assumed that the power consumption at consumers is based on the pricing information, which is a continuously changed variable. The pricing information is sent to consumers (smart meters) by an aggregator via wireless link and the attacker is able to jam the pricing signal within a certain area. During the jamming, the consumers will remain the power consumption amount because they do not have the up-to-date pricing information. When there is a significant change of the pricing information, the attacker stops jamming. The sudden change of the pricing information will cause a significant change on power consumption in a short time, and consequently affects the power grid stability.

In this case we assume that the attacker compromised the load controller (smart meter) 11, 12, 13, and 15, 16, 17, which locate within a nearby area geographically. We also assume the extreme case that during the jamming attack, consumers simply do not consume any power, and then operate under full load when the jamming stops and updated pricing signal is received. As a comparison, we also analyze this scenario and simulate it in single domain using PSCAD.

When being considered in cyber-physical cross domain, however, the single domain scenario setup is over-

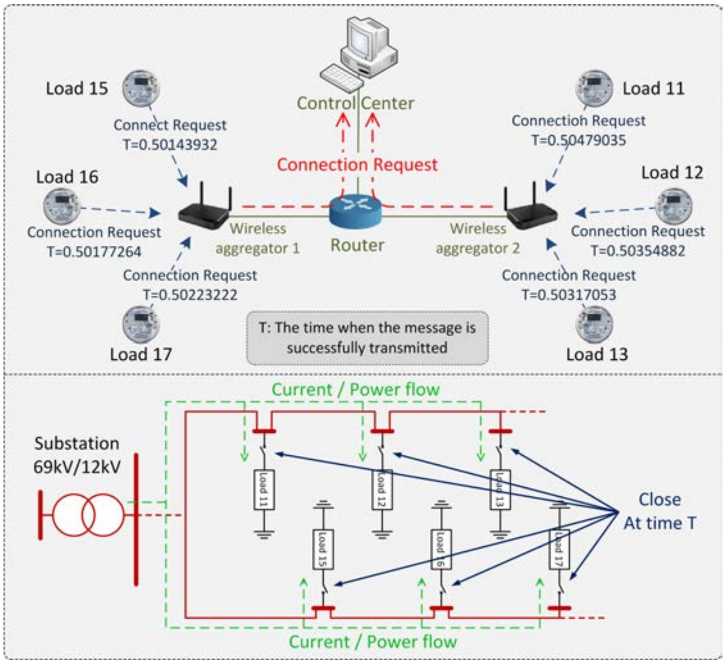


Figure 57: Jamming the Price Signal attack.

idealistic. In practice the smart meters won't be able to communicate with the wireless aggregator exactly at the same time, because wireless channel can only be used by one host at any time. A more realistic simulation is deployed in the *Greenbench*, and the simulation setup is shown in Figure 57. Wireless aggregator 1 (WA1) is the access point for load 15, 16, and 17, while wireless aggregator 2 (WA2) is the access point for load 11, 12, and 13. There is no interference between WA1 and WA2 area, but hosts within each area will contend to access the wireless channel. And the physical load is assumed to be connected to the power system immediately when its load controller gains the access to its WA and its connection request is received by the control center.

Because of wireless channel contention, the connection requests from those 6 loads do not arrive at control center at the same time, and hence the physical loads also take turns to be connected to main power grid. Although the time between each load get connected is very short, it is enough for the power grid to be prepared for the load change, and therefore the current and real power change is much more smooth than that in wireline communications, which indicates the system stability is unlikely to be impacted.

Remark 18. *In this case, the attack causes a real load change, and the attacker's goal is to cause an instability to power system by the sudden load change. A similar attack named "distributed internet-based load altering attack" [123] also follows this type, in which the attacker is assumed gained the control of smart meters over a large area, and by turning off a large amount of household load, e.g., water heater in 1000 homes, the power grid stability is negatively impacted. However, as shown by* Greenbench *simulation, this type of attack actually bears low risk mainly because the contention period of wireless communication acts as a buffer which mitigates the "sudden" change so that the power grid has enough time to prepare for the load change.*

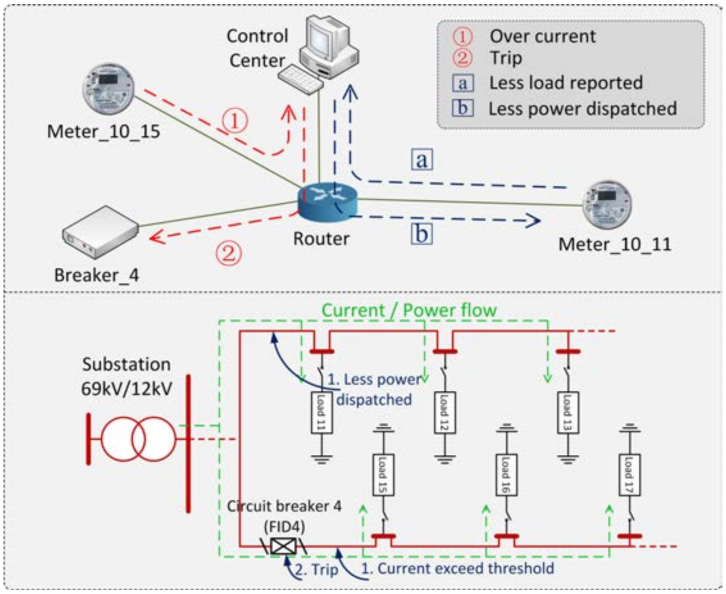
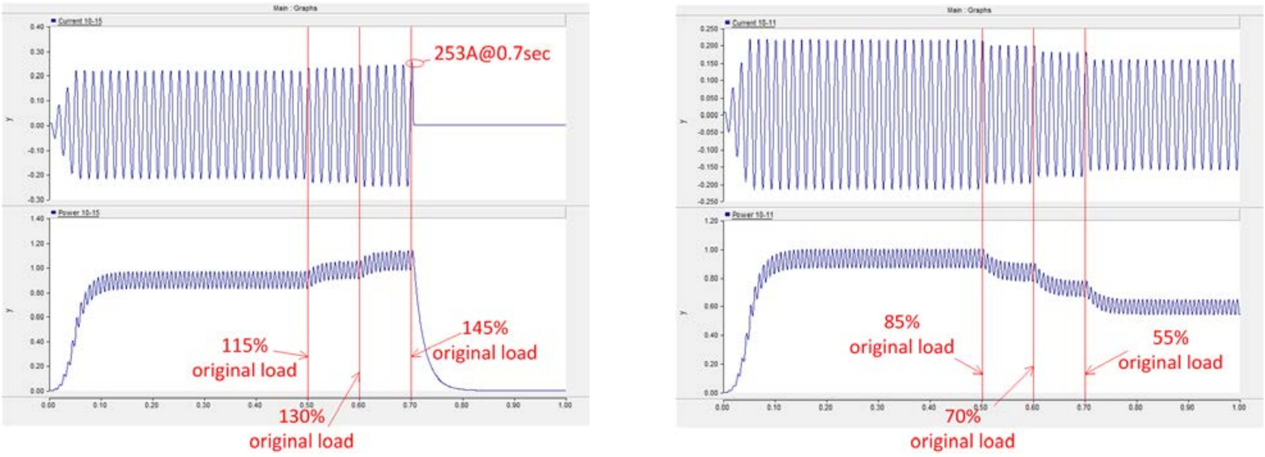


Figure 58: Load Redistribution attack.



(a) Current and power flow through Meter_10_15

(b) Current and power flow through Meter_10_11

Figure 59: Load redistribution attack simulation in *Greenbench*.

Distorted Load Attacks In this case the Load Redistribution (LR) attack [124] is simulated. The LR attack is a special type of the false data injection attack [116]. The false data injection attack refers to an attack in which carefully designed false data could be added to certain group of monitored data in power grid, however, those false data can not be detected by the *state estimation* algorithm which is used to detect bad data in power grid. This false data is accepted and used by the control center to make decision, although it is not in consistent with real device status, and this inconsistency may cause unpredictable damage to power grid.

In the LR attack, the author put some constrains on the attackable nodes in smart grid, which makes LR attack more practical and easier to be launched. Particularly, while in original false data injection attack the author treat each node homogeneously, in LR attack it is assumed that only the load nodes are attackable. Note in this attack, the attacker's goal is not to change the real load – the power consumed by a device, but to modify the load reading, which is the monitored value sent to the control center. And we use *real load* and *load reading* to different the two concepts hereafter.

Same as in case I, we also assume the attacker compromised meters which provide readings for load 11, 12, 13, 15, 16, and 17. Two special constrains of the LR attack are that the overall real load consumption of the attacked area remains the same, while the load reading changes for each specific load does not exceed 50% of its original load. According to these constrains, we setup the attack scenario as following:

1. Assume the attacker increases the load reading at load 15, 16, and 17; and decreases it at load 11, 12, and 13. The total increased load at load 15, 16, 17 and total decreased load at load 11, 12, 13 sum to zero.

2. The attack is launched in 3 time steps with 0.1 second time-interval between each step. For each step, at load 15, 16, and 17, the attacker increases their load reading by 15% of their original load, and at the same time he decreases the same amount of load reading at load 11, 12, and 13. The total load reading change for each load is 45% of its original load.

3. Note that in this simulation, our goal is different from [124]. In [124], the goal of the attack is to find a combination of load redistribution which causes the maximum cost, while our goal is to deploy this attack in a real cyber-physical system and study its potential physical impacts rather than its economic cost. Therefore it is unnecessary to solve the optimization problem used in [124].

The simulation setup is shown in Figure 58. In Figure 58, the Meter_10_15 and Meter_10_11 are meters which monitor the current and power flow on the feeder segment between load 10 - load 15, and load 10 - load 11, and their sample frequency are set to be 10 samples (messages) per second. The maximum threshold on feeders in both section 2 and section 4 is set to be 250A. And in this simulation we collect only the meter reading from Meter_10_15 and Meter_10_11 as it is intuitive that the feeders in segment 10 - 11 and 10 - 15 hold the maximum current and real power value in their own branches, and thus they are the first ones to fail if there is an over-current on these branches. The Breaker_4 represents the circuit breaker controller of FID 4.

The simulation result is shown in Figure 59, and the attack steps are described as below:

1. **t=0.5s:** Attacker launches attack. Both branches operate normally and the current remains at 210A.

2. **t=0.5s-0.7s:** Load reading in section 4 increases with 15% per 0.1 sec, while load reading in section 2 decreases with the same pace.

3. **t=0.7s:** Current at section 4 exceeds threshold by reaching 253A, and over-current message is sent to control center. Control center sends trip message to breaker 4, and section 4 loses power.

On the other hand, as shown in Figure 59(b), because the monitored load decreases in section 2, less power is dispatched to this branch, and consequently the current is much lower than it should be, which will also cause abnormal behavior of power devices in this section.

Remark 19. *In this case, the attack does not change any real load consumption, on the contrary, it modifies the messages sent by meters and aims at confuse the control center of monitored load consumption and real load consumption. As shown by the result, this type of attack is more dangerous. Because the control center is bewildered of the real state, it makes an incorrect decision, which is more harmful than merely a sudden load change.*

Remark 20. Greenbench *simulation of the two cases suggests a smart grid security solution which is instructive for smart grid security research: relatively, a single or a set of smart meters being compromised and gained control does not put smart grid under a high risk; as long as the attacker is unable to forge an authentic message, the whole smart grid is safe. Therefore, compared to fortify smart meter and keep it from being compromised, we should pay more attention on designing security policies to authenticate messages and detect a bad or inconsistent message even if a meter is compromised.*

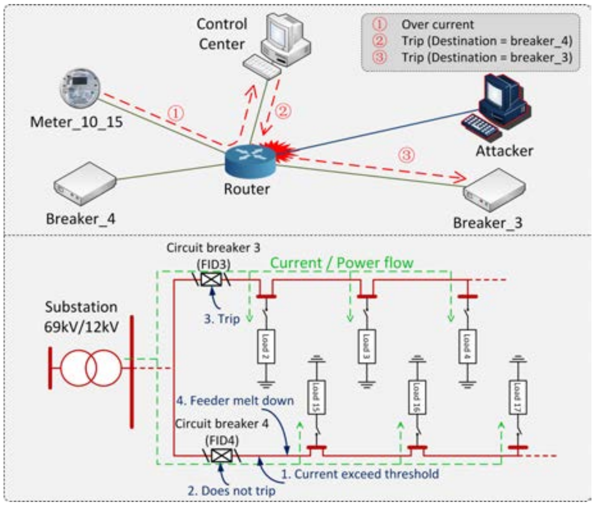
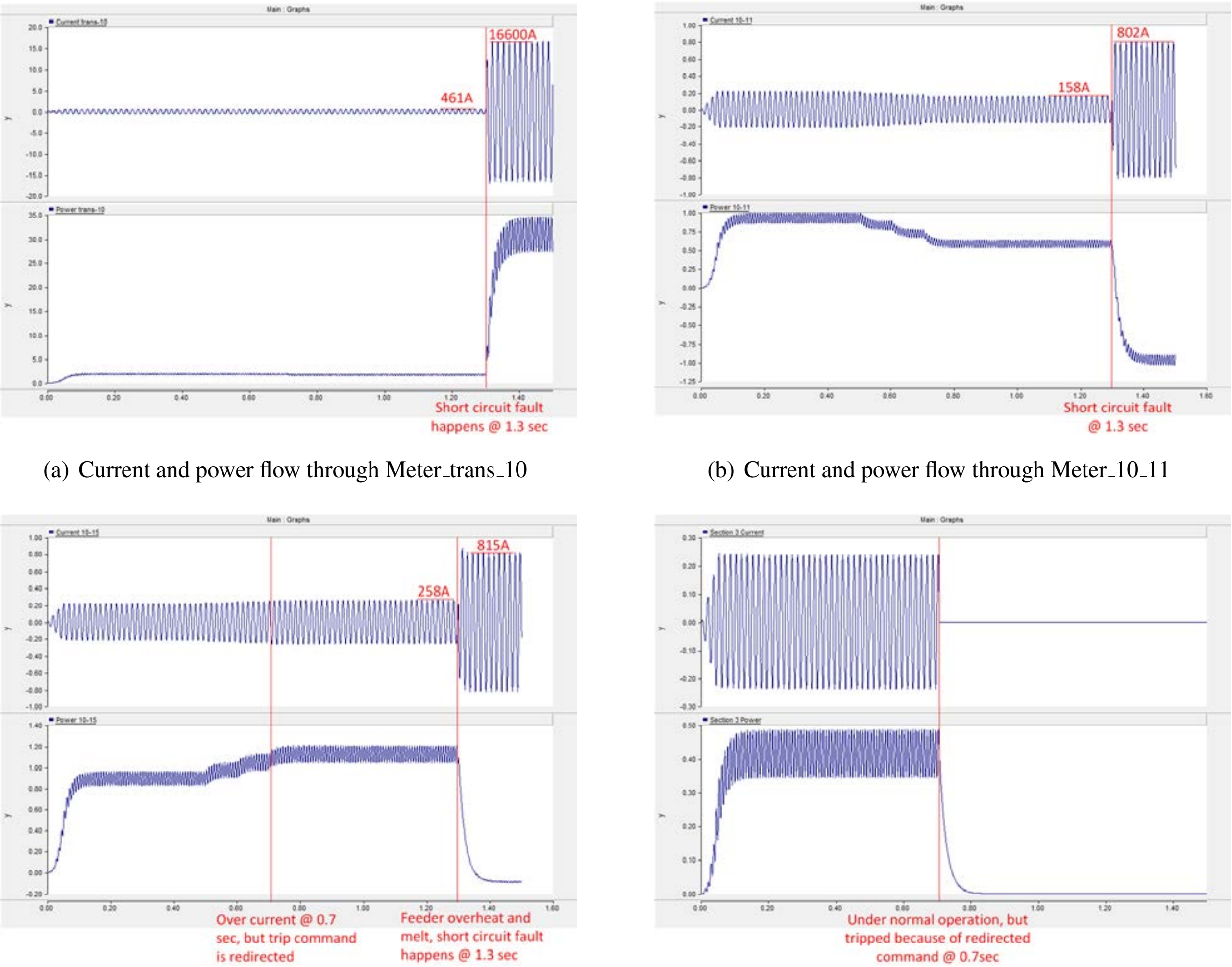


Figure 60: LR attack and Man-in-the-middle attack.

Composite Attacks: Distorted Data and Man-in-the-Middle Attack The power grid is a critical infrastructure and is state-owned in many countries, thus those who sabotage power grid assumes serious crime. It is reasonable to assume the power grid targeted attack is made by clear purpose and therefore the attacker will explore every possibility to maximize the damage. Rather than a single attack, the attacker is highly likely to launch multiple attacks which affects more devices.



(a) Current and power flow through Meter_trans_10

(b) Current and power flow through Meter_10_11

(c) Current and power flow through Meter_10_15

(d) Current and power flow through Meter of section 3

Figure 61: Attack combination simulation in *Greenbench*.

In this case we assume a skilled attacker combines more than one attacks and tries to cause a more severe impact to the smart grid. Specifically, we assume that at the same time an LR attack is launched, the attacker also compromises a router and applies a Man-in-the-middle attack in which he eavesdrops messages processed by the router, locates the "trip" message sent from control center to breaker 4, and modifies the destination address of the "trip" message from breaker 4 to breaker 3. This scenario is shown as in Figure 60.

Figure 61 shows the current and real power values at different devices/points in the Green Hub. For example, we use "trans_10" denote the feeder segment between substation transformer and load 10. Given that an LR attack begins at 0.5 second, we found that at 0.7 second, the monitored current at Meter_10_15 exceeded 250A, and the control center sent the "trip" message to breaker 4. Here we examine the subsequent events as follows:

1. Because the attacker also compromised the router, the "trip" message sent by the control center was modified, and the message destination was changed to breaker 3.

2. As a direct result of the redirected message, breaker 3 trips and causes a blackout of the whole section 3, as shown in Figure 61(d).

3. Also, since breaker 4 did not receive the "trip" message from the control center, the circuit breaker remains closed, which makes the feeder in section 4 run under an over-current state.

4. At time $t = 1.3$ second, which is 0.5 seconds after section 3 running under abnormal condition, the extra heat caused by the over-current causes the feeder to melt and a feeder-to-ground short circuit fault happens.

5. Consequently, we can see the disastrous impact to the whole power grid, as shown in Figure 61(a), Figure 61(b) (current and power flow in Section 2, about 4 times of their normal values), and Figure 61(c) (current and power flow in Section 4).

6. It is observed that the power values on both branches suddenly dropped to negative, which indicates a reverse current flow.

7. Then a more severer damage is triggered on the feeder segment from substation transformer to load 10, which is shown in Figure 61(a). The current on this feeder surged from around 450A to 16,600A, more than 30 times of its normal operate value.

Such a huge serge will surely cause severe damage to the connected power devices. Then the transformers and even the substation are also very likely to be damaged, which may serve as an initial point of a larger-area cascading failure.

Remark 21. *As indicated in the simulation result, the composite attacks cause much severer impact than any of the single attack. This result indicate another non-intuitive solution: although an attack on any single device is unavoidable, its impact could be limited by making it difficult for the attacker to combine various attacks. The most intuitive solution (yet always being neglected in practice) is to use different login/password for different devices. For more sophisticated solutions, one could deploy a hierarchical security policy, in which different levels of devices are protected by different security methods (physically locked and deploying surveillance camera, using encryption algorithms such as AES, etc).*

6.2.4 Summary

We developed *Greenbench*, a cross-domain simulation platform which could capture the impact of cyber attacks in power systems. Along with *Greenbench*, We study the data-centric attacks which target at damaging power grid by manipulating the data exchanged between devices. The simulation results convey non-intuitive indications and instructive suggestions to both smart grid security research and deployment. Nonetheless, *Greenbench's* capability is not confined to this, its flexibility and extensibility allows us to analysis and evaluate various smart grid attacks, which is one of our future works. As another future work, we will integrate power grid dedicated communication protocols such as DNP3 and IEC-61850, and evaluate attacks targeting those protocols.

6.3 Relevance to Original Goals

To facilitate classification of networks and identify anomalous networks, we introduced a generalized Markov Graph model and its application to social network classification and social network synthesis. The main result of the generalized Markov Graph model is that the degree distribution, the clustering coefficient distribution, and the crowding coefficients are three fundamental statistics for characterizing generic social networks. In addition, the generalized Markov Graph model provides a new insight into clustering coefficient: it is the result of the dependence between higher order structures, namely the triads, in social networks.

To detect and track systematic failures in networks, We have presented a distributed algorithm to track a systematic failure in sensor networks which is robust to random failures. The algorithm presented has a low communication complexity, and requires a small number of local communications which makes it ideal for real time applications. We assume neither any global coordinate information nor any capability of directly sensing the phenomenon causing the failure. We presented precise mathematical formulations of the algorithms along with proofs for their correctness. The simulations performed demonstrate the accuracy and robustness of our methods.

Our proposed framework, *Greenbench*, makes new contributions in two-fold. First of all, *Greenbench* is a cross-domain simulation platform that includes an underlying power system *overlayed* by a communication system. In such a way, we are able to capture the impact of cyber attacks in power systems in real-time, unlike networking simulations as [125] or physical systems like [126, 127, 128]. Second, we aim to study the consequences of *data-centric attacks* rather than manipulation of communication protocols like [129]. The benefits of our efforts is that there might be many attacks or manipulation schemes to attack smart grid, however, the resulting of attacks and countermeasures for data integrity will be revealed in ultimate data received, which could be *delayed* and/or *distorted*. To this end, our study is able to demonstrate the direct impact of security attacks at the power system level, either due to compromised smart meters in AMI or DoS attacks to messages in transmission.

7 Broad Impact, Workforce Training, and Dissemination

WMD can only have an impact because it is a human initiative. This, in turn, is of course a result of some issue a group of individuals have taken up as a a rallying point to operate for spreading fear and terror, and destruction of infrastructure. Studying the underlying social stratum which may lead to such abject plotting and planning or even discussing a WMD scenario hence makes sense as a holistic approach. In particular, in this work, we have explored fundamental and robust models of social networks which can allow us to detect specific structures. As we will elaborate below, we have proposed a new model beyond the ”Preferential Attachment”, so that we allow a very robust comparison and classification of social networks, as a first step of detecting suspicious activity planning. We have also developed a mathematical framework which can model a ”propagation of belief” in a network, and a capacity to influence this propagation, and the trend of the belief.

7.1 Contribution to the Body of Knowledge

Our work focused on the impact, detection, and tracking of WMDs on networks. The modern infrastructure relies heavily on many types of networks, and their failures may be heavily consequential. In the course of our work over the last five years, our formulation of effective strategies for timely detection and tracking of such attacks on networks has led to several journal and conference publications listed in the references, as well as several invited talks, including, Imperial College, University of Luxemburg, Univ. of Illinois-UC, University of Minnesota and the Institute of Mathematics and its Applications, University of Canberra, Australia, and Chalmers University in Sweden. Dr. Krim has also been voted on the editorial Board of the Flagship of the Signal Processing Magazine, where he was invited to write a feature tutorial article on the mathematical and algorithmic tools developed in the course of this DTRA-funded research. Dr. Krim has in addition, thanks to the partial support of DTRA, submitted the first draft on an upcoming book on “Geometric Methods in Signal and Image Analysis with Cambridge Press”.

7.2 Personnel Support

Two faculty members, two post-doc, and a total of five PhD students at North Carolina State University are supported during this study:

- Dr. Wenye Wang, Associate Professor in the Department of Electrical and Computer Engineering.
- Dr. Hamid Krim, Professor in the Department of Electrical and Computer Engineering.
- Dr. Fei Xing, a former Ph.D. student in the Department of Electrical and Computer Engineering. Graduated in January 2010.
- Ming Zhao, Ph.D., under the supervision of Dr. Wenye Wang, December 2009
- Dr. Yi Xu, Research associate in the Department of Electrical and Computer Engineering. Graduated in May 2010 under the supervision of Dr. Wenye Wang.
- Dr. Lei Sun, a former Ph.D. student in the Department of Electrical and Computer Engineering under the supervision of Dr. Wenye Wang.
- Dr. Harish Chintakunta, a former student in the Department of Electrical and Computer Engineering under the supervision of Dr. Hamid Krim.

- Dr. Tian Wang, a former student in the Department of Electrical Engineering and graduated in Theoretical Physics under the supervision of Dr. Hamid Krim

- Dr. Jennifer Gamble, partially supported in the Department of Electrical Engineering and graduated under the supervision of Dr. Hamid krim

- Zhuo Lu, Ph.D. student in the Department of Electrical and Computer Engineering under the supervision of Dr. Wenye Wang.

7.3 Social Dimension of WMD Attacks

7.3.1 Background and Rationale

As an abstract model of a social environment, a social network includes a set of nodes, which could be a set of individuals or a set of groups of individuals, and a set of relationships among these nodes. The analysis of social networks is important in several aspects. For example, it reflects characteristics of a social environment, so that we can recognize important or interesting nodes, groups of nodes or even societies. It can also help us understand how a social environment evolves so that we are able to make predictions about its impact. Moreover, we might even learn how to control a social environment based on the prediction and recognition techniques to benefit the people in the environment.

Social networks have been studied for decades, and several simple but fundamental models have been proposed. For social network synthesis, the Barabasi-Albert model [130] successfully generates social networks which follow a power law degree distribution based on the assumption of preferential attachment. In statistics, the Markov Graph model[131], the p* model [132] as well as the models from Statistical Mechanics [133] are well known for characterizing the probability distribution of networks. Other social network models were also proposed in engineering [134, 135]. More recently, we proposed a method based on a Markov Graph model to address the problem of social network classification [136].

The characteristics of nodes, to help identify key nodes in a social network, are of particular interest to researchers. The classification of social networks, however, has hardly been addressed despite its great importance in applications. For example, in criminal networks, the potential use of social network classification would be to detect whether different criminal networks belong to a larger network, hence using similar tactics. In terrorist networks, this method could, for instance, be used to detect whether a terrorist network is led by the same leader who has a history of organizing other known terrorist plots, and hence to help identify the leader.

7.3.2 Our Contributions

We were the first to propose a method based on a Markov Graph model to classify different types of social networks [136]. The classification method makes use of two features, the degree list and the number of triads, to determine the probability distribution of networks in a Markov Graph model and to use them as the crucial features for classification of social networks. As we demonstrate later, this simple model is insufficient to provide the classification performance improvement which we seek.

In the research area of social network synthesis, Barabasi-Albert model as well as many other related models enable the synthesis of the evolution of simple social networks whose degree distributions obey the power law. The algorithms for these models are clear and efficient. The open question, however, and in light of these models' shortfall of fully capturing the probabilistic structure of a social network, is about the statistical behavior of the required additional features, namely the clustering and crowding coefficients, where the former reveals the information between a node's neighbours, and the latter describes the neighboring environment of a triad of people in a social network.

To this end, we proposed a model which provides answers to both of these questions. In some sense, the preferential attachment algorithm by Barabasi-Albert is closely related to the simple Markov Graph model. In the preferential attachment algorithm, the probability of a new edge attaching to a node depends on the number of edges attached to the node, which is the degree of the node. And in a Markov Graph model, the basic assumption is that the probability of a new edge being formed, depends on a function whose variables are the states of the relationships between the nodes in this new edge and all of the other nodes.

Towards addressing all these questions, we proposed a so called generalized Markov Graph model [137]. The characteristics of this new model include the dependences on the relationships between pairs of nodes in the network, as well as the relationships between triplets of nodes. On this basis, we also build a new algorithm for social network synthesis. The main result of the model is shown next :

Theorem 22. *Any simple network graph G with an associated dependence graph D_{GM} has a probability mass function:*

$$\begin{aligned} P(G) =& z^{-1} exp(\sum_n s_n(G)\theta_n + \sum_i c_i(G)\gamma_i \\ &+ \sum_j t_j(G)\tau_j), \end{aligned} \tag{72}$$

where z is a normalization factor, $s_n(G)$ stands for the number of n^{th} k-stars in network G, θ_n is the associated coefficient, $c_i(G)$ is the number of i^{th} cluster-stars, γ_i is the associated coefficient, $t_j(G)$ is the number of j^{th} tri-stars and τ_j is the associated coefficient. Subscripts n, i and j count from the first to the last structures they correspond to.

In light of this new model, we have discovered that the probability distribution of generic social networks will depend on not only the degree list, but also on the clustering coefficient list as well as the crowding coefficient list. These features can subsequently be applied to the classification of social networks, as well as to the validation of the generalized preferential attachment algorithm.

To further validate the models and reveal more potential applications of the models, we test the probability density function of the elements in the adjacency matrix as a direct test of the basic assumptions of the model. Such a result is then directly applied to a belief control mechanism in information flow models [138, 139]. We first propose an information flow model (IFM) of belief that captures how interactions among members affect the diffusion and eventual convergence of a belief. The IFM model reveals that the diffusion of beliefs heavily depends on two characteristics of the social network structure, namely degree centralities and clustering coefficients. We apply IFM to both analyze and control the convergence of a belief. We capture the structure of the social network using two different techniques, namely the preferential attachment and the generalized Markov Graph model. We evaluate our models via experiments with published real social network data. The flow chart of the IMF model is showed as below.

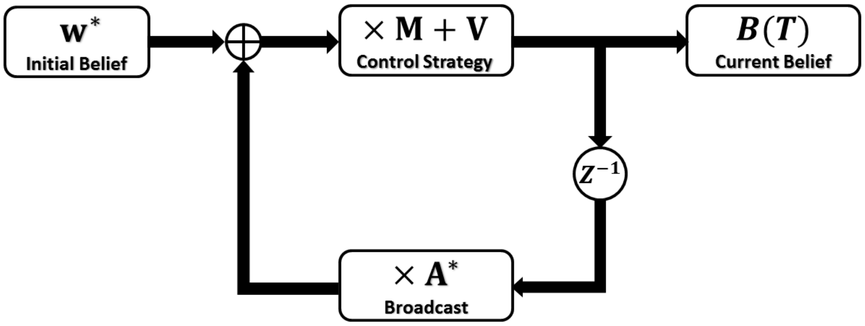


Figure 62: Flow Chart for Information Flow Model.

With the real network data, the experimental results show that the generalized Markov Graph model correctly generated the converged beliefs in information flow models. In addition, the control strategy of beliefs derived from a generalized Markov Graph model outperformed the state-of-art social network models.

We also explored the relationship between a variation on a generalized Markov Graph model (GMGV) and high dimensional Laplace operators, which are also called the edge Laplacians. The reason these two concepts are related is that they both are based on the idea that networks are partitioned into different units, i.e. simplices, and the dependence graph of the GMGV is very closely related with the Laplacian operator. In addition, we have observed that the eigen-space of a Laplacian operator has very unique properties, including integer eigenvalues and corresponding sparse eigenvectors. The GMGV could give a reasonable explanation to such phenomena because cliques in its dependence graph do correspond to the non-zero elements in the sparse eigenvectors with integer eigenvalues. This is also a novel way of identifying high dense subgroups with symmetry in a social network.

7.3.3 Summary

In summary, we introduced a generalized Markov Graph model. The main result of the model is that the degree distribution, the clustering coefficient distribution, and the crowding coefficients are shown to be three fundamental statistics for characterizing generic social networks. In order to validate the assumptions of the model, experiments on the probability distribution of all adjacency matrix elements are performed. In addition, classification experiments are carried out to show that the three resulting statistics in the model do effectively characterize social networks. To further verify the model as well as show its potential application, we apply the generalized Markov Graph model in the information flow problems. All experimental results and applications show that the generalized Markov Graph model outperformed the state-of-art models. To further explore the model and give a direction in the future research, we investigated the relationship between a variation on a generalized Markov Graph model and the eigen-space of Laplacians of a network.

7.4 Potential Applications

The new works are proposed to enhance and extend our earlier efforts in the past three years. In the scope of the original project, we focused on modeling node behaviors, such as being cooperative, faulty, destructive, and dead, and analyzing network responses to a multitude of node failures, in terms of network survivability and connectivity, and design of protocols and algorithms for network robustness. While, in this extended period, we are still pursuing the fundamental study of network responses to WMD attacks, we, however, aim to explore the impact of failures in temporal and spatial domains, e.g., when a failure occurs, how far-away nodes will be affected, and how long it takes to notice such a failure, instead of just probabilistic estimation and connectivity analysis. The proposed works are clearly more challenging, given the limited existing theoretical knowledge of these issues. As a result, our efforts will advance the knowledge and fundamental understanding of the of WMD attacks, and the resulting failures in the infrastructure networks which form the backbone of civilian and military capabilities. Further, our results, we believe, will provide significant insights on the scope of damage due to cascading and correlated failures, including catastrophic loss of connectivity, unsuccessful missions, slow-down of the Internet, and damage to the economy and society at large. All together, the proposed research will leverage DoD capability in response to attacks from WMD/WME.

Failure Prevention via Inter-Cooperation: For a large-scale network, a more and more promising solution is to enable inter-operation of communication systems using different access technologies, including wired and wireless, on licensed and unlicensed frequency bands. The objectives can be two-fold in general: improving bandwidth utilization and improving communication capacity. While the former one is more for offering better services to civilian world, the latter one is more for offering *critical* communications in emergency, disaster rescure, and even opportunistic communications. Therefore, our results provide insights on two fronts. First, we have found that the latency for failure in communication networks may propagate with an upper bound, depending on the *scales* of the network, while there exists a lower bound for the information to be delivered to other nodes. Second, our results suggest useful parameters, such as node density, mobility range, and transmission power etc toward design objectives, respectively.

Failure Containment: Our results on the spreading of correlated failures provide the conditions to contain failures. We can hence sample and test a subset of nodes in the network infrastructure to evaluate the network robustness through a detailed examination on the failure correlations among these nodes. If the failure correlations are in the percolation regime, we are alarmed to take actions to reduce the failure dependence in the network and hence improve the network resilience. Our findings are also useful at the network planning and construction phases by providing the failure resistance guidelines.

Radio Resource Recycling: Our analysis of the performance limits in cognitive radio networks demonstrates the feasibility of information delivery through recycling the temporarily unused radio resources in the recovery network. Since the recovery network is limited in the availability of radio resources, the cognitive radio technology is important to supplement the scheduling schemes to make the full network utilization. Our results further provide the essential information in dimensioning the cognitive radio networks that achieve the information delivery goals. Additionally, our findings provide the theoretical foundation for the cognitive radio network deployment in multiple phases and predict the network performance in accordance to each deployment phase.

Network Synthesis and Failure Detection: Significant damage to network infrastructures will cause devastating consequences and may cascade onto many other systems. Crucial information distributed in a surveillance network may be lost. In our work, we develop a distributed and localized algorithm to accurately detect and track systematic failures in sensor networks which may indicate the deployment of WMD in the region the network is deployed. By distributed, we mean, we do not gather the data at any central processing location, and by localized, we mean that only the information of local neighborhood is sufficient for a node to determine if it is on the front of a propagating systematic failure. Over the past year, we have made significant progress in precisely characterizing networks in general probabilistically and have run extensive experimentation. A journal paper is under review and a conference paper has been published. We have also made significant progress in WMD impact on failing sensor networks with localization, evolution of failures and their complete characterizations, together with techniques to mitigate and overcome failed regions.

Failures and Vulnerability in Physical Networks: Smart grid is an emerging cyber-physical system which is expected to replace traditional power grid in near future. Traditional power grid has been running for decades without significant changes on its infrastructure and begins to show its inability as the demand for power delivery and consumption boosts in recent years. One main reason which causes the inefficiency of traditional power grid is the lack of a full-fledged communication infrastructure. Although there exists a control and monitor network which is built above the traditional power grid, most power devices still operate in an isolated manner and their operation is based on electrical properties rather than information exchange. For example, a relay makes the decision to open a circuit breaker only when it detects the current on a feeder exceeds the threshold, it neither tells other relays its own status nor takes information from other relays to help itself make a decision. The lack of information exchange makes traditional power grid fragile because in many situations it is too late to take action when there is a noticeable physical change. The integration of communication networks with power grid, however, brings a new host of vulnerabilities stem from Internet and opens the door for potential adversaries to tear down a physical system through a cyber attack, which is clearly relevant to DTRA missions.

7.5 Participation and Presentations

Our recent presentations include:

- Harish Chintakunta and Hamid Krim, "Distributed boundary tracking using alpha and delaunay-cech shapes," demonstrated at the 17th International Conference on Discrete Geometry for Computer Imagery (DGCI), 2013.
- Harish Chintakunta and Hamid Krim, "A Distributed Collapse of a Network's Dimensionality," IEEE Global Conference on Signal and Information Processing, pp. IPN.PB.4, 2013.
- Chintakunta, H.; Krim, H. , "Detection and tracking of systematic time-evolving failures in sensor net-

works," Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2011 4th IEEE International Workshop on , vol., no., pp.373-376, 13-16 Dec. 2011 doi: 10.1109/CAMSAP.2011.6136029

- Gamble, J.; Chintakunta, H.; Krim, H.;, "Applied topology in static and dynamic sensor networks," International Conference on Signal Processing and Communication (SPCOM), 2012.
- Wang, T.; Krim, H.;, "Statistical Classification of Social Networks", IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2012.
- Lei Sun and Wenye Wang, "On Latency Distribution and Scaling: From Finite to Large Cognitive Radio Networks under General Mobility," presented by Lei Sun at IEEE INFOCOM 2012, Orlando, April 2012.
- Lei Sun and Wenye Wang, "Understanding the Tempo-spatial Limits of Information Dissemination in Multi-channel Cognitive Radio Networks," presented by Lei Sun at IEEE INFOCOM 2012, Orlando, April 2012.
- Yi Xu and Wenye Wang, "Scheduling Partition for Order Optimal Capacity in Large-Scale Wireless Networks," presented by Yi Xu at ACM MobiCom 2009, Beijing, China, September 2009.
- Yi Xu and Wenye Wang, "Characterizing the Spread of Correlated Failures in Large Wireless Networks," presented by Yi Xu at IEEE INFOCOM 2010, San Diego, CA, March 2010.
- Lei Sun and Wenye Wang, "On Study of Achievable Capacity with Hybrid Relay in Cognitive Radio Networks," presented by Lei Sun at IEEE GLOBECOM 2009, Honolulu, Hawaii, November 2009.
- Harish Chintakunta and Hamid Krim, "Divide and Conquer: Localizing Coverage Holes in Sensor Networks," presented by Harish Chintakunta at IEEE SECON 2010, Boston, MA, June 2010.

References

[1] J. Yoon, M. Liu, and B. Noble, “Sound mobility models,” in *Proc. of ACM MobiCom*, (San Francisco, CA, USA), pp. 205–216, September 2003.

[2] Y. Fang, I. Chlamtac, and Y.-B. Lin, “Portable movement modeling for PCS networks,” *IEEE Transactions on Vehicular Technology*, vol. 46, pp. 1356–1363, July 2000.

[3] G. Wang, G. Cao, and T. L. Porta, “Movement-assisted sensor deployment,” in *Proc. of IEEE INFOCOM*, vol. 4, (Hong Kong, China), pp. 2469–2479, March 2004.

[4] N. Bansal and Z. Liu, “Capacity, delay and mobility in wireless ad-hoc networks,” in *Proc. of IEEE INFOCOM*, (San Francisco, CA, USA), pp. 1553–1563, April 2003.

[5] C. Bettstetter, “On the connectivity of ad hoc networks,” *Computer Journal, Special Issue on Mobile and Pervasive Computing*, no. 4, pp. 432–447, July 2004.

[6] P. Samar and S. B. Wicker, “On the Behavior of Communication Links of Node in a Multi-Hop Mobile Environment,” in *Proc. of ACM MobiHoc ’04*, May 2004.

[7] P. Nain, D. Towsley, B. Liu, and Z. Liu, “Properties of Random Direction Models,” in *Proc. of IEEE Infocom ’05*, Mar. 2005.

[8] M. McGuire, “Stationary distribution of random walk mobility models for wireless ad hoc networks,” in *Proc. of ACM MobiHoc*, (Urbana-Champaign, IL, USA), pp. 90–98, May 2005.

[9] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” in *Mobile Computing* (Imielinski and Korth, eds.), vol. 353, ch. 5, pp. 153–181, Kluwer Academic Publishers, 1996.

[10] C. Bettstetter, “Smooth is Better than Sharp: A Random Mobility Model for Simulation of Wireless Networks,” in *Proc. of ACM MSWiM*, 2001.

[11] B. Liang and Z. Haas, “Predictive Distance-Based Mobility Management for PCS Networks,” in *Proc. of IEEE INFOCOM*, April 1999.

[12] D. Heyman and M. Sobel, *Stochastic Models in Operations research.* McGraw-Hill, 1982.

[13] M. Zhao and W. Wang, “A Unified Mobility Model for Analysis and Simulation of Mobile Wireless Networks,” *ACM-Springer Wireless Networks (WINET)*, vol. 15, pp. 365–389, April 2009.

[14] J. Yoon, M. Liu, and B. Noble, “Random Waypoint Considered Harmful,” in *Proc. of IEEE INFOCOM*, 2003.

[15] C. Bettstetter, G. Resta, and P. Santi, “The node distribution of the random waypoint mobility model for wireless ad hoc networks,” *IEEE Transactions on Mobile Computing*, vol. 2, pp. 257–269, July - September 2003.

[16] G. Carlsson, V. de Silva, and D. Morozov, “Zigzag persistent homology and real-valued functions,” in *Proceedings of the 25th annual symposium on Computational geometry*, pp. 247–256, ACM, 2009.

[17] Y. Peres, A. Sinclair, P. Sousi, and A. Stauffer, “Mobile geometric graphs: Detection, coverage and percolation,” in *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 412–428, SIAM, 2011.

[18] B. Liu, O. Dousse, P. Nain, and D. Towsley, “Dynamic coverage of mobile sensor networks,” *Parallel and Distributed Systems, IEEE Transactions on*, vol. 24, no. 2, pp. 301–311, 2013.

[19] J. Gamble, H. Chintakunta, and H. Krim, “Applied topology in static and dynamic sensor networks,” in *Signal Processing and Communications (SPCOM), 2012 International Conference on*, pp. 1–5, IEEE, 2012.

[20] V. Srinivasan, P. Nuggehalli, C. F. Chiasserini, and R. R. Rao, “Cooperation in Wireless Ad Hoc Networks,” in *Proc. of IEEE Infocom ’03.*, pp. 808 – 817, 30 March-3 April 2003.

[21] I. Aad, J.-P. Hubaux, and E. W. Knightly, “Denial of Service Resilience in Ad Hoc Networks,” in *Proc. of ACM MobiCom ’04*, 2004.

[22] D. M. Nicol, W. H. Sanders, and K. S. Trivedi, “Model-Based Evaluation: From Dependability to Security,” *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 1, pp. 48–65, 2004.

[23] M. L. Shooman, *Reliability of Computer Systems and Networks*. New York: John Wiley & Sons, INC, 2004.

[24] G. Weichenberg, V. Chan, and M. Medard, “High-Reliability Architectures for Networks under Stress,” in *Proc. of IEEE INFOCOM’04*, vol. 1, pp. 131 – 141, March 2004.

[25] B. Bollobas, *Modern Graph Theory*. Springer, 1998.

[26] C. Bettstetter, “On the Connectivity of Ad Hoc Networks,” *The Computer Journal, Special Issue on Mobile and Pervasive Computing*, vol. 47, no. 4, pp. 432–447, 2004.

[27] D. Miorandi and E. AltmanA, “Coverage and Connectivity of Ad Hoc Networks in Presence of Channel Randomness,” in *Proc. of IEEE Infocom ’05*, pp. 491–502, Mar. 2005.

[28] S. Ross, *Stochastic Processes*. John Wiley and Sons Inc., 2nd ed., 1996.

[29] N. Limnios and G. Oprisan, *Semi-Markov Processes and Reliabiltiy*. Boston: Birkha äuser,c/o Springer-Verlag, 2001.

[30] D. Moore, G. Voelker, and S. Savage, “Inferring Internet Denial-of-Service Activity,” in *Proc. of the 2001 USENIX Security Symposium (Best Paper Award)*, 2001.

[31] L. Buttyan and J.-P. Hubaux, “Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks,” *Mobile Networks and Applications*, vol. 8, no. 5, pp. 579–592, Oct. 2003.

[32] F. Stevens, T. Courtney, S. Singh, A. Agbaria, J. F. Meyer, W. H. Sanders, and P. Pal, “Model-Based Validation of an Intrusion-Tolerant Information System,” in *Proc. of 23rd IEEE International Symposium on Reliable Distributed Systems (SRDS’04)*, pp. 184–194, Oct. 2004.

[33] M. Franceschetti, L. Booth, M. Cook, R. Meester, and J. Bruck, “Percolation of multi-hop wireless networks,” Tech. Rep. UCB/ERL M03/18, EECS Department, University of California, Berkeley, 2003.

[34] W. Mendenhall and T. Sincich, *”Statistics for Engineering and the Sciences, 4th Edition ”*. Englewood Cliffs, NJ: Prentice Hall, 1995.

[35] G. Corradi, J. Janssen, and R. Manca, “Numerical Treatment of Homogeneous Semi-Markov Processes in Transient Case – a Straightforward Approach,” *Methodology and Computing in Applied Probability*, vol. 6, pp. 233–246, 2004.

[36] J.-K. Lee and J. C. Hou, “Modeling Steady-state and Transient Behaviors of User Mobility: Formulation, Analysis, and Application,” in *Proc. of ACM MobiHoc ’06*, pp. 85–96, May 2006.

[37] F. Xing and W. Wang, “Modeling and Analysis of Connectivity in Mobile Ad Hoc Networks with Misbehaving Nodes,” in *Proc. of IEEE ICC’06*, June 2006.

[38] M. Zhao and W. Wang, “Design and Applications of A Smooth Mobility Model for Mobile Ad Hoc Networks,” in *Proc. of IEEE Milcom’06*, October 2006.

[39] L.Anderegg and S. Eidenbenz, “Ad hoc-VCG: a Truthful and Cost-Efficient Routing Protocol for Mobile Ad Hoc Networks with Selfish Agents,” in *Proc. of ACM MobiCom’03*, pp. 245–259, September 2003.

[40] L. Buttyan and J.-P. Hubaux, “Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks,” *Mobile Networks and Applications*, pp. 579–292, February 2003.

[41] Z. Nikoloski1, N. Deo, and L. Kucera, “ Degree-correlation of a Scale-free Random Graph Process,” in *Proc. of European conference on Combinatorics, Graph Theory and Applications (Eurocomb’05)*, pp. 239–244, 2005.

[42] M. D. Penrose, “On k-connectivity for a geometric random graph,” *Random Struct. Algorithms*, vol. 15, no. 2, pp. 145–164, 1999.

[43] M. Penrose, *Random Geometric Graphs*. Oxford University Press, 2003.

[44] R. Hekmat, *Ad-hoc Networks: Fundamental Properties and Network Topologies*. Springer Netherlands, 1 ed., 2006.

[45] C. Bettstetter, “On the Minimum Node Degree and Connectivity of a Wireless Multihop Network,” in *Proc. of the ACM MobiHoc ’02*, pp. 80–91, ACM Press, June 9-11 2002.

[46] U. of Bonn, “BonnMotion: A Mobility Scenario Generation and Analysis Tool, available at *http://web.informatik.uni-bonn.de/IV/Mitarbeiter/dewaal/BonnMotion/*,” 2005.

[47] D. Blough, M. Leoncini, G. Resta, and P. Santi, “The k-Neighbors Approach to Physical Degree Bounded and Symmetric Topology Control in Ad Hoc Networks ,” *IEEE Transactions on Mobile Computing*, vol. 5, pp. 1267–1282, Sep. 2006.

[48] O. Dousse, F. Baccelli, and P. Thiran, “Impact of interferences on connectivity in ad hoc networks,” *IEEE/ACM Transactions on Networking*, vol. 13, no. 2, pp. 425–436, 2005.

[49] F. Xue and P. Kumar, “The Number of Neighbors Needed for Connectivity of Wireless Networks,” *Kluwer Wireless Networks*, vol. 10, pp. 169–181, Mar. 2004.

[50] P. Balister, B. Bollobas, A. Sarkar, and M. Walters, “Connectivity of Random K-nearest-neighbour Graphs,” *Advances in Applied Probability*, vol. 37, no. 1, pp. 1–24, 2005.

[51] B. Bollobas and O. Riordan, *Percolation*. Cambridge University Press, 2006.

[52] R. Zheng, “Information Dissemination in Power-Constrained Wireless Networks,” in *Proc. of IEEE Infocom’07*, April 2006.

[53] C. Yi and W. Wang, “On the Connectivity Analysis over Large-Scale Hybrid Wireless Networks ,” in *Proc. of IEEE INFOCOM)*, April 2010.

[54] Y. Xu and W. Wang, “The Speed of Information Propagation in Large Wireless Networks,” in *Proceedings of IEEE INFOCOM*, April 2008.

[55] O. Dousse, P. Mannersalo, and P. Thiran, “Latency of Wireless Sensor Networks with Uncoordinated Power Saving Mechanisms,” in *Proceedings of ACM MOBIHOC*, May 2004.

[56] Z. Kong and E. M. Yeh, “On the Latency for Information Dissemination in Mobile Wireless Networks,” in *Proceedings of ACM MOBIHOC*, May 2008.

[57] M. Grossglauser and D. Tse, “Mobility increases the capacity of ad hoc wireless networks,” in *IEEE Transactions on Networking*, vol. 10, pp. 477–486, August 2002.

[58] M. J. Neely and E. Modiano, “Capacity and Delay Tradeoffs for Ad-Hoc Mobile Networks,” in *IEEE Transactions on Information Theory*, vol. 51, June 2005.

[59] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, “Throughput-Delay Trade-off in Wireless Networks,” in *Proceedings of IEEE INFOCOM*, Apr 2004.

[60] S. Toumpis and A. Goldsmith, “Large Wireless Networks under Fading, Mobility, and Delay Constraints,” in *Proceedings of IEEE INFOCOM*, Apr 2004.

[61] S. Toumpis and A. Goldsmith, “Delay and Capacity Trade-offs in Mobile Ad Hoc Networks: A Global Perspective,” in *Proceedings of IEEE INFOCOM*, Apr 2006.

[62] A. A. Hanbali, A.A.Kherani, R. Groenevelt, P. Nain, and E. Altman, “Impact of Mobility on the Performance of Relaying in Ad Hoc Networks,” in *Proceedings of IEEE INFOCOM*, Apr 2006.

[63] R. Moraes, H. Sadjadpour, and J. G.-L. Aceves, “Mobility-Capacity-Delay Trade-off in Wireless Ad Hoc Networks,” in *Elsevier Journal on ad hoc Networks*, July 2005.

[64] M. Garetto, P. Giaccone, and E. Leonardi, “Capacity Scaling in Delay Tolerant Networks with Heterogeneous Mobile Nodes,” in *Proceedings of ACM MOBIHOC*, May 2007.

[65] R. Zheng, “Information Dissemination in Power-Constrained Wireless Networks,” in *Proceedings of IEEE INFOCOM*, April 2006.

[66] W. Ren, Q. Zhao, and A. Swami, “Connectivity of Cognitive Radio Networks: Proximity vs. Opportunity,” in *Proceedings of ACM MOBICOM Workshop on Cognitive Radio Networks*, September 2009.

[67] P. Gupta and P. R. Kumar, “The Capacity of Wireless Networks,” in *IEEE Transactions on Information Theory*, vol. 46, pp. 388–404, March 2000.

[68] L. Sun and W. Wang, “On Latency Distribution and Scaling: From Finite to Large Cognitive Radio Networks under General Mobility,” in *Proc. of IEEE INFOCOM’12)*, April 2012.

[69] T. Liggett, “An Improved Subadditive Ergodic Theorem,” in *The Annals of Probability*, vol. 13, pp. 1279–1285, 1985.

[70] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci., “Wireless sensor networks: a survey,” *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[71] A. Hatcher, *Algebraic Topology.* Cambridge University Press, 2002.

[72] D. Donoho and C. Grimes, “ When does isomap recover natural parameterization of families of articulated images,” Tech. Rep. 2002-27, Stanford University, August 2002.

[73] J. Tanenbaum, S. Mika, B. Schlkopf, and R. Williamson, “ Regularized principal manifolds,” *Journal of Machine Learning Research,*, vol. 1, pp. 179–209, June 2001.

[74] H. Chintakunta and H. Krim, “Distributed localization of coverage holes using topological persistence,” *Signal Processing, IEEE Transactions on*, vol. 62, pp. 2531–2541, May 2014.

[75] A. Tahbaz-Salehi and A. Jadbabaie, “Distributed coverage verification in sensor networks without location information,” *Automatic Control, IEEE Transactions on*, vol. 55, no. 8, pp. 1837–1849, 2010.

[76] H. Chintakunta and H. Krim, “Divide and conquer: Localizing coverage holes in sensor networks,” in *Sensor Mesh and Ad Hoc Communications and Networks (SECON), 2010 7th Annual IEEE Communications Society Conference on*, pp. 1 –8, june 2010.

[77] J. Bruck, J. Gao, and A. A. Jiang, “Localization and routing in sensor networks by local angle information,” *ACM Transactions on Sensor Networks*, vol. 5, no. 1, 2009.

[78] N. Kimura and S. Latifi, “A survey on wireless multimedia sensor networks,” *Computer Networks*, vol. 51, no. 4, pp. 921–960, 2007.

[79] J. W. Vick, *Homology theory : an introduction to algebraic topology*. springer, 1994.

[80] A. hatcher, *Algebraic Topology*. Cambridge University Pr, 2002.

[81] V. de Silva and R. Ghrist, "Coordinate-free coverage in sensor networks with controlled boundaries via homology," *The International Journal of Robotics Research*, vol. 25, no. 12, pp. 1205–1222, 2006.

[82] A. Muhammad and M. Egerstedt, "Control using higher order Laplacians in network topologies," *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, pp. 1024–1038, 2006.

[83] A. Muhammad and A. Jadbabaie, "Decentralized computation of homology groups in networks by gossip," *Proceedings of American Control Conference*, pp. 3438–3443, 2007.

[84] A. Tahbaz-Salehi and A. Jabdabaie, "Distributed Coverage Verification in Sensor Networks Without Location Information," *proceedings of the 47th IEEE conference on Decision and Control*, pp. 4170 – 4176, 2008.

[85] M. Khabbazian, H. Mercier, and V. K. Bhargava, "Wormhole attack in wireless ad hoc networks: analysis and countermeasure," *Proceedings of the Global Telecommunications Conference (GLOBECOM)*, 2006.

[86] L. Hu and D. Evans, "Using Directional Antennas to Prevent Wormhole Attack," *Proceedings of Network and Distributed System Security Symposium (NDSS)*, 2004.

[87] H. K. Chintakunta, *Topology and Geometry of Sensor Networks: A Distributed Computing Approach*. PhD thesis, North Carolina State University, 2013.

[88] A. Sen, S. Murthy, and S. Banerjee, "Region-based connectivity: a new paradigm for design of fault-tolerant networks," in *Proceedings of the 15th international conference on High Performance Switching and Routing*, HPSR'09, (Piscataway, NJ, USA), pp. 94–100, IEEE Press, 2009.

[89] B. Hao, A. Sen, and B. H. Shen, "A new metric for fault-tolerance in sensor networks," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, SenSys '04, (New York, NY, USA), pp. 289–290, ACM, 2004.

[90] D. Arifler, "Information theoretic approach to detecting systematic node destructions in wireless sensor networks," *Wireless Communications, IEEE Transactions on*, vol. 7, pp. 4730 –4738, november 2008.

[91] C. Farah, C. Zhong, M. Worboys, and S. Nittel, "Detecting topological change using a wireless sensor network," in *Proceedings of the 5th international conference on Geographic Information Science*, GIScience '08, (Berlin, Heidelberg), pp. 55–69, Springer-Verlag, 2008.

[92] C. Jiang, G. Dong, and B. Wang, "Detection and tracking of region-based evolving targets in sensor networks," in *Computer Communications and Networks, 2005. ICCCN 2005. Proceedings. 14th International Conference on*, pp. 563 – 568, oct. 2005.

[93] J. Jiang and M. Worboys, "Detecting basic topological changes in sensor networks by local aggregation," in *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, GIS '08, (New York, NY, USA), pp. 4:1–4:10, ACM, 2008.

[94] H. Chintakunta and H. Krim, "Divide and conquer: Localizing coverage holes in sensor networks," in *Sensor Mesh and Ad Hoc Communications and Networks (SECON), 2010 7th Annual IEEE Communications Society Conference on*, pp. 1 –8, june 2010.

[95] A. Muhammad and A. Jadbabaie, "Decentralized computation of homology groups in networks by gossip," in *American Control Conference, 2007. ACC '07*, pp. 3438 –3443, july 2007.

[96] S. Duttagupta, K. Ramamritham, and P. Kulkarni, "Tracking dynamic boundaries using sensor network," *Parallel and Distributed Systems, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2011.

[97] A. Yezzi, L. Zllei, and T. Kapur, "A variational framework for joint segmentation and registration," in *IN IEEE MATHEMATICAL METHODS IN BIOMEDICAL IMAGE ANALYSIS*, pp. 44–51, 2001.

[98] N. Papadakis and E. Mémin, "A variational technique for time consistent tracking of curves and motion," *J. Math. Imaging Vis.*, vol. 31, pp. 81–103, May 2008.

[99] E. Huot, H. Yahia, and I. Herlin, "Landslide tracking with a curve evolution model driven by interferometric data," in *Geoscience and Remote Sensing Symposium, 2003. IGARSS '03. Proceedings. 2003 IEEE International*, vol. 6, pp. 3802 – 3804 vol.6, july 2003.

[100] Y. Rathi, N. Vaswani, A. Tannenbaum, and A. Yezzi, "Tracking deforming objects using particle filtering for geometric active contours," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, pp. 1470 –1475, aug. 2007.

[101] R. Albert and A. Barabási, "Statistical mechanics of complex networks," *REVIEWS OF MODERN PHYSICS*, vol. 74, pp. 47–96, JANUARY 2002.

[102] O. Frank and D. Strauss, "Markov graphs," *Journal of the American Statistical Association*, vol. 81, pp. 832–842, Sep 1986.

[103] M. E. J. Newman, "Proc. natl. acad. sci. usa 98, 404-409." http://www-personal.umich.edu/ mejn/netdata/, 2001.

[104] M. Mihail and N. K. Vishnoi, "On generating graphs with prescribed vertex degrees for complex network modeling," *ARACNE*, pp. 1–11, 2002.

[105] J. Park and M. E. J. Newman, "Statistical mechanics of networks," *PHYSICAL REVIEW E*, vol. 70, 2004.

[106] J. Bachrach and C. Taylor, *Localization in Sensor Networks*, pp. 277–310. John Wiley & Sons, Inc., 2005.

[107] J. Aspnes, T. Eren, D. Goldenberg, A. Morse, W. Whiteley, Y. Yang, B. Anderson, and P. Belhumeur, "A theory of network localization," *Mobile Computing, IEEE Transactions on*, vol. 5, pp. 1663 –1678, dec. 2006.

[108] D. Moore, J. Leonard, D. Rus, and S. Teller, “Robust distributed network localization with noisy range measurements,” in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, SenSys ’04, (New York, NY, USA), pp. 50–61, ACM, 2004.

[109] J. Gao, L. J. Guibas, J. Hershberger, L. Zhang, and A. Zhu, “Geometric spanner for routing in mobile networks,” in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, MobiHoc ’01, (New York, NY, USA), pp. 45–55, ACM, 2001.

[110] C. Avin, *Random Geometric Graphs: An Algorithmic Perspective*. PhD thesis, 2006.

[111] H. Chintakunta and H. Krim, “Detection and tracking of systematic time-evolving failures in sensor networks,” in *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2011 4th IEEE International Workshop on*, pp. 373 –376, dec. 2011.

[112] H. Chintakunta and H. Krim, “Distributed boundary tracking using alpha and delaunay-Čech shapes,” demonstrated at International Conference On Discrete Geometry for Computer Imagery, 2013.

[113] H. Chintakunta and H. Krim, “Distributed boundary tracking using alpha and delaunay-Čech shapes,” availabe online at `http://arxiv.org/pdf/1302.3982.pdf`, 2013.

[114] L. Pietre-Cambacedes, C. Chalhoub, and F. Cleveland, “Iec tc57 wg15–cyber security standards for the power systems,” *CIGR É Study Committee D*, vol. 2, 2007.

[115] D. Watts, “Security and vulnerability in electric power systems,” in *35th North American power symposium*, vol. 2, pp. 559–566, 2003.

[116] Y. Liu, P. Ning, and M. K. Reiter, “False data injection attacks against state estimation in electric power grids,” in *16th ACM Conference on Computer and Communication Security*, (NY, USA), pp. 21–32, 2009.

[117] “W32.Stuxnet Dossier.” `www.symantec.com`.

[118] H. Hooshyar, “System protection for high pv-penetrated residential distribution systems (green hubs).,” 2011.

[119] H. Li and Z. Han, “Manipulating the electricity power market via jamming the price signaling in smart grid,” in *GLOBECOM Workshops (GC Wkshps), 2011 IEEE*, pp. 1168–1172, IEEE, 2011.

[120] C. Efthymiou and G. Kalogridis, “Smart grid privacy via anonymization of smart metering data,” in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pp. 238–243.

[121] T. Yardley, R. Berthier, D. Nicol, and W. H. Sanders, “Smart grid protocol testing through cyber-physical testbeds,” in *Innovative Smart Grid Technologies (ISGT), 2013 IEEE PES*, pp. 1–6, IEEE, 2013.

[122] S. McLaughlin, D. Podkuiko, and P. McDaniel, “Energy theft in the advanced metering infrastructure,” in *Critical Information Infrastructures Security*, pp. 176–187, Springer, 2010.

[123] A.-H. Mohsenian-Rad and A. Leon-Garcia, “Distributed internet-based load altering attacks against smart power grids,” *Smart Grid, IEEE Transactions on*, vol. 2, no. 4, pp. 667–674, 2011.

[124] Y. Yuan, Z. Li, and K. Ren, “Modeling load redistribution attacks in power systems,” *Smart Grid, IEEE Transactions on*, vol. 2, no. 2, pp. 382–390, 2011.

[125] A. AlMajali, A. Viswanathan, and C. Neuman, “Analyzing resiliency of the smart grid communication architectures under cyber attack,” in *5th Workshop on Cyber Security Experimentation and Test*, 2012.

[126] D. Kundur, X. Feng, S. Mashayekh, S. Liu, T. Zourntos, and K. Butler-Purry, “Towards modelling the impact of cyber attacks on a smart grid,” *Intl Journal of Security and Networks*, vol. 6, no. 1, pp. 2–13, 2011.

[127] H. Lin, S. Sambamoorthy, S. Shukla, J. Thorp, and L. Mili, “Power system and communication network co-simulation for smart grid applications,” in *Innovative Smart Grid Technologies (ISGT), 2011 IEEE PES*, pp. 1–6, IEEE, 2011.

[128] K. Hopkinson, X. Wang, R. Giovanini, J. Thorp, K. Birman, and D. Coury, “Epochs: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components,” *Power Systems, IEEE Transactions on*, vol. 21, no. 2, pp. 548–558, 2006.

[129] J. Mirkovic, T. V. Benzel, T. Faber, R. Braden, J. T. Wroclawski, and S. Schwab, “The deter project: Advancing the science of cyber security experimentation and test,” in *Technologies for Homeland Security (HST), 2010 IEEE International Conference on*, pp. 1–7, IEEE, 2010.

[130] R. Albert and A. Barabasi., “Statistical mechanics of complex networks.,” *REVIEWS OF MODERN PHYSICS*, vol. 74, no. 1, pp. 47–96,, 2002.

[131] O. Frank and D. Strauss., “Markov graphs,” *Journal of the American Statistical Association*, vol. 81, no. 9, pp. 832–842, 1986.

[132] S. Wasserman and P. Patrison., “Logit models and logistic regressions forsocial networks: I. an inttroduction to markov graphs and p*,” *PSYCHOMETRIKA*, vol. 61, no. 3, pp. 401–425, 1996.

[133] J. Park and M. E. J. Newman., “Statistical mechanics of networks,” *PHYSICAL REVIEW E*, vol. 70, 2004.

[134] V. M. Preciado and A. Jadbabaie., “Moment-based spectral analysis of large-scale networks using local structural information,” *IEEE/ACM Transactions on Networking*, vol. 99, no. 1, 2012.

[135] S. Hanneke, W. Fu, and E. P. Xing., “Discrete temporal models of social networks,” *Electron. J. Statist.*, vol. 4, pp. 585–606, 2010.

[136] T. Wang and H. Krim, “Statistical classiffication of social networks,” in *EEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3977–3980, March 2012.

[137] H. Y. V. Wang, Tian; Krim, “A generalized Markov Graph model: Application to social network analysis,,” *IEEE Journal of Selected Topics in Signal Processing,*, vol. 7, no. 2, pp. 318–332, 2013.

[138] H. Wang, Tian; Krim, “Control and prediction of beliefs on social network,” in *The Fifth IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2013.

[139] H. Y. V. Wang, Tian; Krim, “Analysis and Control of Beliefs in Social Networks,” *IEEE Transaction on Signal Processing*, 2015 (to appear).

DEPARTMENT OF DEFENSE

DEFENSE THREAT REDUCTION
AGENCY
8725 JOHN J. KINGMAN ROAD
STOP 6201
FORT BELVOIR, VA 22060
ATTN: P. TANDY

DEFENSE TECHNICAL
INFORMATION CENTER
8725 JOHN J. KINGMAN ROAD,
SUITE 0944
FT. BELVOIR, VA 22060-6201
ATTN: DTIC/OCA

DEPARTMENT OF DEFENSE CONTRACTORS

QUANTERION SOLUTIONS, INC.
1680 TEXAS STREET, SE
KIRTLAND AFB, NM 87117-5669
ATTN: DTRIAC